



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH  
TECHNOLOGIÍ

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF CONTROL AND INSTRUMENTATION

## DOCHÁZKOVÝ SYSTÉM S WWW SERVEREM

MICROCONTROLLER SYSTEM FOR OFFICE ACCESS MONITORING

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. MARTIN TRÍSKA

VEDOUCÍ PRÁCE

SUPERVISOR

doc. Ing. ZDENĚK BRADÁČ, Ph.D.

BRNO 2013



VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

Ústav automatizace a měřicí techniky

# Diplomová práce

magisterský navazující studijní obor  
**Kybernetika, automatizace a měření**

**Student:** Bc. Martin Tříška

**ID:** 115300

**Ročník:** 2

**Akademický rok:** 2012/2013

## NÁZEV TÉMATU:

**Docházkový systém s WWW serverem**

## POKYNY PRO VYPRACOVÁNÍ:

Navrhněte koncepci docházkového systému vybaveného WWW rozhraním. Systém navrhněte jako miniaturní systém vybavený mikrokontrolérem a sadou V/V. Navrhněte připojení senzorů a akčních členů a oživte. Vybavte programovým vybavením, otestujte a předved'te funkčnost. Zaměřte se na maximální cenovou efektivitu.

## DOPORUČENÁ LITERATURA:

Pavel Herout: Učebnice jazyka C, KOPP, 2004, IV. přepracované vydání, ISBN 80-7232-220-6  
Dle pokynů vedoucího práce.

**Termín zadání:** 11.2.2013

**Termín odevzdání:** 20.5.2013

**Vedoucí práce:** doc. Ing. Zdeněk Bradáč, Ph.D.

**Konzultanti diplomové práce:**

**doc. Ing. Václav Jirsík, CSc.**

*Předseda oborové rady*

## UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## **Abstrakt**

Diplomová práce se zabývá návrhem a realizací terminálu pro docházkový systém. Úvod stručně prezentuje několik řešení, která jsou v současnosti dostupná na trhu. Hlavní část práce se pak dělí na dva celky, z nichž první popisuje návrh hardwarové platformy a druhý tvorbu softwarové výbavy pro vyvíjené zařízení. Terminál je realizován jako kompaktní s širokými možnostmi budoucího rozšíření. Důraz je kladen krom univerzálnosti také na uživatelskou přívětivost.

## **Klíčová slova**

Docházka, ARM, STM32, Cortex M3, FriendlyARM, S3C2440, Linux, webový server

## **Abstract**

This thesis describes the design and realization of terminal for time attendance system. The introductory part shortly presents several solutions currently available on the market. The main part of this thesis is divided into two sections. The first of them deals with design of hardware platform, and the second describes the development of software. The terminal is realized as a compact with wide possibilities for future functional expansion. Special attention is given to the flexibility and user-friendly interface.

## **Keywords**

Time attendance, STM32, ARM, Cortex M3, FriendlyARM, S3C2440, Linux, web server

### **Bibliografická citace:**

TŘÍSKA, M. *Docházkový systém s WWW serverem*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2013. 73s. Vedoucí diplomové práce byl doc. Ing. Zdeněk Bradáč, Ph.D.



## **Prohlášení**

„Prohlašuji, že svou diplomovou práci na téma Docházkový systém s WWW serverem jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne: **20. května 2013**

.....  
podpis autora

## **Poděkování**

Děkuji vedoucímu diplomové práce doc. Ing. Zdeňku Bradáčovi, Ph.D. za odborné, metodické a pedagogické rady, týkající se této práce a také za jeho snahu vyjít vždy vstříc. Děkuji také společnosti Tipa spol. s r.o. Opava za materiální podporu projektu a Ing. Lukáši Kopečnému, Ph.D. za ochotu při zprostředkování CNC obrábění. Největší dík si však zaslouží rodina za skvělou podporu po celou dobu mého studia.

V Brně dne: **20. května 2013**

.....  
podpis autora

# Obsah

1	Úvod.....	9
2	Existující řešení.....	10
2.1	Evidenční docházkový systém Fingera.....	10
2.1.1	Terminály.....	10
2.1.2	Koncepce systému .....	11
2.2	ACS line – Elektronický identifikační systém.....	12
2.2.1	Terminály.....	12
2.2.2	Software .....	13
2.3	Miaxis biometrics – Attendance management .....	13
2.3.1	Terminály.....	13
2.3.2	Software .....	14
3	Koncepce systému .....	15
3.1	Typické nasazení.....	15
3.2	Požadavky .....	15
4	Hardware terminálu .....	17
4.1	Kit MINI2440.....	17
4.2	Rozšiřující deska .....	18
4.2.1	Napájení .....	19
4.2.2	Mikrokontrolér.....	20
4.2.3	Binární vstupy a výstupy .....	22
4.2.4	Audio zesilovač.....	23
4.2.5	Nabíječ Li-Ion akumulátoru.....	25
4.2.6	DC/DC měnič záložního zdroje.....	26
4.2.7	Měření napětí .....	28
4.2.8	Řadič kapacitní klávesnice.....	29
4.2.9	Čtečka RFID karet .....	30
4.2.10	Ostatní rozhraní.....	31
4.2.11	Shrnutí.....	32
4.3	Čtečka otisků prstu .....	33
4.4	Ostatní prvky .....	35

4.5	Pouzdro terminálu .....	36
4.6	Změny v HW části .....	37
5	Softwarová výbava .....	38
5.1	Rozšiřující deska .....	39
5.1.1	Obsluha čtečky otisků .....	40
5.1.2	Obsluha kapacitní klávesnice .....	42
5.1.3	Komunikace s nadřazeným systémem .....	44
5.2	Aplikační server .....	47
5.2.1	Funkcionalita .....	48
5.3	Databáze .....	50
5.3.1	Struktura .....	50
5.3.2	Výkon .....	52
5.4	Vizualizace pro LCD displej .....	53
5.4.1	Přímý přístup k LCD .....	53
5.4.2	Prostředí Qt .....	53
5.4.3	Funkce .....	54
5.5	Webové rozhraní .....	57
5.5.1	JQuery .....	58
5.5.2	Struktura .....	59
5.5.3	Použití webového rozhraní .....	59
5.6	Ostatní části SW výbavy .....	62
5.6.1	Statsgen .....	62
5.6.2	Skripty .....	63
5.6.3	Převzaté programy .....	64
5.7	Spotřeba energie .....	64
6	Závěr .....	66

# 1 ÚVOD

Celosvětový trend honby za materiálním bohatstvím – ziskem je hlavním hnacím motorem ekonomického systému většiny vyspělých zemí – kapitalismu. Vyššího zisku z vyráběného či dodávaného produktu lze, jak vyplývá z jeho definice, dosáhnout buď zvýšením prodejní ceny produktu, nebo snížením nákladů na jeho výrobu. Možnost zdražení však nemusí být vzhledem ke konkurenci v daném oboru a situaci na trhu výhodná, a tak přichází na řadu varianta druhá – snižování nákladů. Jedná se o velmi často diskutované téma, jelikož ho lze dosáhnout mnoha odlišnými způsoby. Patří mezi ně i používání méně kvalitních materiálů a postupů, majících za následek zhoršení užitné hodnoty a (často záměrné) zkracování životnosti výrobku. V kontextu této práce je pro nás však zajímavý jiný přístup ke snižování nákladů. Je jím efektivní využívání lidských zdrojů. Konkrétně nastavená politika v této oblasti je záležitostí přístupu dané společnosti (zaměstnavatele). Důležitým společným faktorem je však u značné části z nich kvalitní evidence pracovní doby, která výrazně napomáhá kvantitativně vyjádřit důležitý ekonomický ukazatel – produktivitu práce. A právě nástrojem, usnadňujícím evidenci pracovní doby se zabývá tato diplomová práce.

Nástroje, používané pro tento účel v minulosti, mezi které patřily tzv. „píchačky“ či někde se stále vyskytující sešit a tužka, nemohou svou uživatelskou přívětivostí a komfortem při zpracování dat konkurovat prostředkům moderní techniky. Řešení předkládané v této práci si klade za cíl realizovat terminál pro docházkový systém, určený především pro malé podniky. Jako hlavní prostředek pro identifikaci uživatelů bude využívat biometrický údaj – otisk prstu. Dále je během návrhu věnována zvýšená pozornost možnostem dalšího rozšiřování a modifikací po SW i HW stránce dle konkrétních potřeb uživatelů. Ty jsou totiž silně závislé na dané aplikaci a mohou tak být značně různorodé. Proto je nutné především softwarovou část popsanou v této práci chápat pouze jako demonstrační ukázkou, jejímž cílem není v žádném případě pokrytí všech možností, které HW platforma nabízí.

Hlavní část této diplomové práce je členěna do 4 kapitol. V první z nich je stručně představeno několik řešení, která jsou v současné době dostupná na trhu. Druhá část se věnuje návrhu celkové koncepce docházkového systému, stanovení požadavků a volbě klíčových komponent použitých pro stavbu. Ve třetí části je rozebrán návrh jednotlivých elektronických celků, jež jsou v terminálu obsaženy, a též je popsáno mechanické řešení. Čtvrtý celek se pak věnuje jednotlivým prvkům ukázkové softwarové výbavy.

## 2 EXISTUJÍCÍ ŘEŠENÍ

Cílem této kapitoly je stručně představit několik komerčně dostupných systémů, poskytujících podobnou funkcionalitu jako řešení navrhované v této práci. Musí nabízet především biometrickou identifikaci pomocí otisku prstu a obsahovat rozhraní ethernet. Jelikož je nabídka na trhu v této oblasti velice široká, jedná se vzhledem k omezenému rozsahu této práce pouze o velice malý vzorek, sloužící pro ilustraci.

### 2.1 Evidenční docházkový systém Fingera

Jde o českou pobočku společnosti dodávající na trh produkty v oblasti přístupových a docházkových systémů, která se prezentuje cca 250 instalacemi v České a Slovenské republice. V první polovině roku 2013 došlo k zásadní obměně nabídky a k inovaci webové prezentace společnosti. Výrobky dodávané touto firmou se v některých vlastnostech velice podobají terminálu, jehož vývojem se zabývá práce, kterou právě čtete. Obsah následujících podkapitol týkajících se systému Fingera je citován z [1].

#### 2.1.1 Terminály

Fingera terminál je hlavní zařízení využívané pro sledování docházky. Může být používán jako samostatný terminál nebo v případě potřeby více terminálů se připojuje na server. Pomocí tohoto terminálu je možné i otevírat dveře.



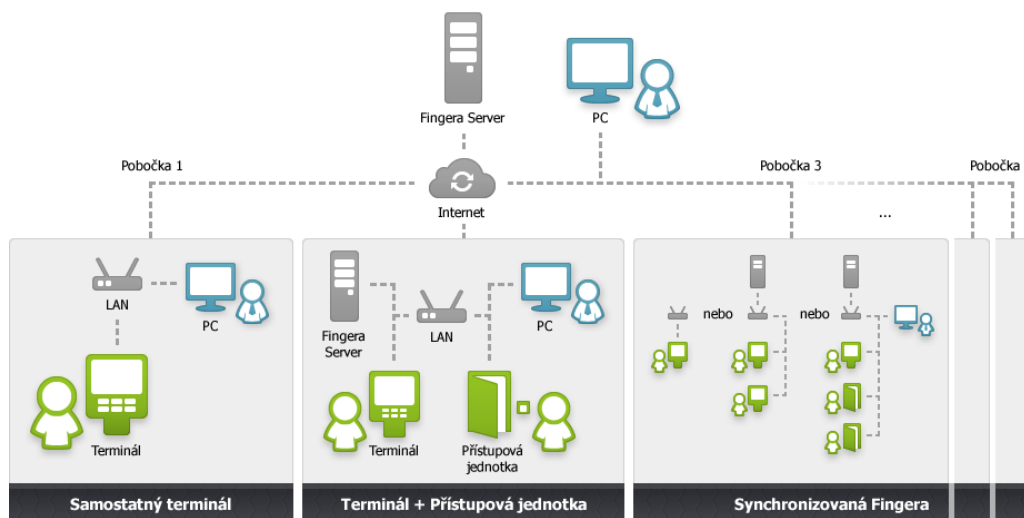
Obrázek 2–1: Terminál systému Fingera [1]

- **Procesor:** ARM8 Cortex vysokorychlostní procesor
- **Displej:** 8" dotykový LCD Display, 800x600
- **Snímač otisků:** Kvalitní optický snímač
- **Snímač karet:** Integrovaný snímač na RFID karty

- **Připojení:** Ethernet 10/100 Mbps
- **USB módy:** USB Host a Client
- **Rychlost rozpoznávání:** Čas identifikace  $\leq 1$  sekunda
- **Audio:** Vestavěné reproduktory
- **Rozměry:** 252(V) x 231(Š) x 53(H) mm
- **Baterie:** Vestavěná vysokokapacitní Li-ion baterie 2000mAh, autonomie do 2 hodin
- **Provozní podmínky:** 0 – 42 stupňů, vlhkost 10% – 80%
- **Provozní napájení:** 12V DC, 1.5A, dodávaný elektrický adaptér
- **Cena:** Docházkový systém je dodáván jako služba, cena se tedy odvíjí od konkrétních podmínek a požadavků aplikace. Pro firmu s cca 80 zaměstnanci a jedním terminálem bez dodatečné funkcionality se cena pohybuje od **1500 € do 2500 €** (od 37 500 Kč do 62 500 Kč při kurzu 1 € / 25 Kč).

### 2.1.2 Koncepce systému

FINGERA je evidenční docházkový systém na bázi otisků prstů nebo RFID bezkontaktních karet určený zejména pro malé a střední podniky, které nejsou spokojeny s evidencí docházky zaměstnanců papírovou formou. Fingera byla vyvinuta společností Innovatrics, což zaručuje vysokou kvalitu rozpoznávání otisků prstů i celého produktu. Využívání biometrie prakticky vylučuje podvody při zapisování časů příchodů a odchodů, čímž zvyšuje disciplínu zaměstnanců a produktivitu práce.



Obrázek 2–2: Možná architektura systému Fingera [1]

## 2.2 ACS line – Elektronický identifikační systém

Jedná se dalšího českého výrobce a dodavatele identifikačních systémů. Jeho nabídka je poměrně široká a obsahuje kompletní řešení pro firmy, jídelny, hotely apod. Pro ilustraci byl zvolen model terminálu, využívající identifikaci pomocí otisků prstů s konektivitou prostřednictvím sítě ethernet. V nabídce se nachází pod označením FT-500F TCP. Materiály následujících podkapitol týkající se systému ACS line jsou čerpány z [2].

### 2.2.1 Terminály

- Grafický display 168 x 64 bodů
- Různobarevné podsvícení
- Možnost úprav zobrazení
- Výpis informací o odpracované době
- 14 univerzálních tlačítek operací
- Oddělená numerická klávesnice
- Integrovaný snímač ID
- Port pro druhý snímač ID
- Možnost zadání PIN
- Duální provedení otisk + karta
- Hlavní komunikace pomocí LAN, rozhraní RS485 pro připojení dalších komponentů
- Napájecí napětí 12 V, spotřeba typicky 270 mA
- Paměť 10 000 záznamů
- vstupy (galvanicky oddělené), 2 výstupy (relé 2 A)
- Cena 23 700Kč za samostatný terminál, 24 999 Kč sada obsahující zjednodušený software a instalační materiál (ceny bez DPH).

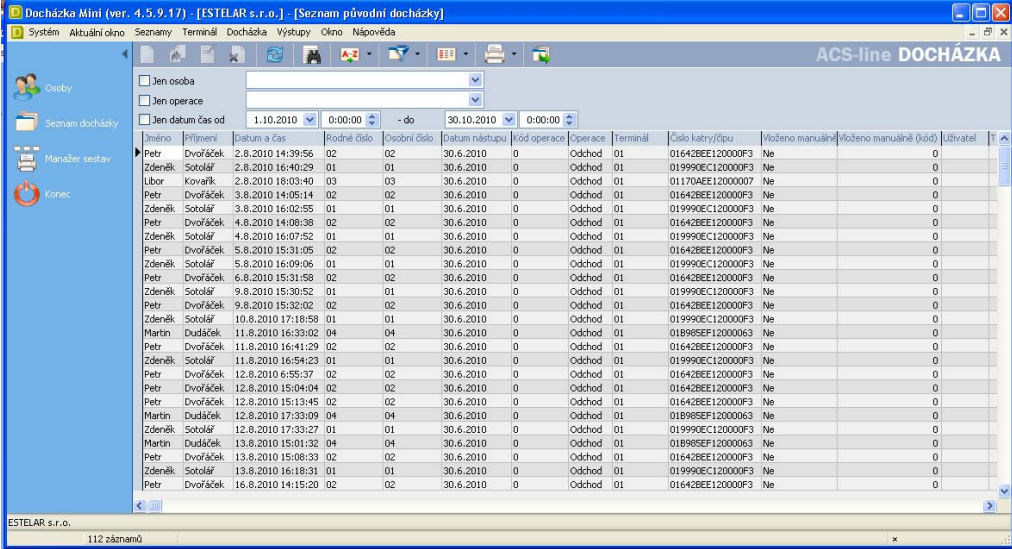


Obrázek 2–3 - Terminál FT-500F TCP výrobce ACS line [2]



## 2.2.2 Software

Pro aplikace, kde nevyhoví zjednodušený software Docházka MINI dodávaný v balíčcích spolu s terminály, je určen program ADS4. Obě aplikace jsou vytvořeny pro operační systém Windows.



The screenshot shows the 'Docházka Mini' software interface. The title bar indicates the version is 4.5.9.17 and it is from ESTELAR s.r.o. The menu bar includes 'Systém', 'Aktuální okno', 'Seznamy', 'Terminál', 'Docházka', 'Výstupy', 'Okno', and 'Nápověda'. The main window has a sidebar with 'Osoby', 'Seznam docházek', 'Manažer sestav', and 'Konec'. The main area displays a table of attendance records with columns: Jméno, Příjmení, Datum a čas, Rodné číslo, Osobní číslo, Datum nástupu, Kód operace, Operace, Terminál, Číslo karty/číslo, Vloženo manuálně, Vloženo manuálně (kód), and Uživatel. The table contains multiple rows of data for various employees like Petr Dvořáček, Zdeněk Sotolář, Libor Kovarik, and others, with dates ranging from 2.8.2010 to 16.8.2010.

Jméno	Příjmení	Datum a čas	Rodné číslo	Osobní číslo	Datum nástupu	Kód operace	Operace	Terminál	Číslo karty/číslo	Vloženo manuálně	Vloženo manuálně (kód)	Uživatel
Petr	Dvořáček	2.8.2010 14:39:56	02	02	30.6.2010	0	Odhod	01	01642BEE120000F3	Ne		0
Zdeněk	Sotolář	2.8.2010 16:40:29	01	01	30.6.2010	0	Odhod	01	019990EC120000F3	Ne		0
Libor	Kovarik	2.8.2010 18:03:40	03	03	30.6.2010	0	Odhod	01	01170AEE12000007	Ne		0
Petr	Dvořáček	3.8.2010 14:05:14	02	02	30.6.2010	0	Odhod	01	01642BEE120000F3	Ne		0
Zdeněk	Sotolář	3.8.2010 16:02:55	01	01	30.6.2010	0	Odhod	01	019990EC120000F3	Ne		0
Petr	Dvořáček	4.8.2010 14:08:38	02	02	30.6.2010	0	Odhod	01	01642BEE120000F3	Ne		0
Zdeněk	Sotolář	4.8.2010 16:07:52	01	01	30.6.2010	0	Odhod	01	019990EC120000F3	Ne		0
Petr	Dvořáček	5.8.2010 15:31:05	02	02	30.6.2010	0	Odhod	01	01642BEE120000F3	Ne		0
Zdeněk	Sotolář	5.8.2010 16:09:06	01	01	30.6.2010	0	Odhod	01	019990EC120000F3	Ne		0
Petr	Dvořáček	6.8.2010 15:31:58	02	02	30.6.2010	0	Odhod	01	01642BEE120000F3	Ne		0
Zdeněk	Sotolář	9.8.2010 15:30:52	01	01	30.6.2010	0	Odhod	01	019990EC120000F3	Ne		0
Petr	Dvořáček	9.8.2010 15:32:02	02	02	30.6.2010	0	Odhod	01	01642BEE120000F3	Ne		0
Zdeněk	Sotolář	10.8.2010 17:18:58	01	01	30.6.2010	0	Odhod	01	019990EC120000F3	Ne		0
Martin	Dudáček	11.8.2010 16:33:02	04	04	30.6.2010	0	Odhod	01	018985EF12000063	Ne		0
Petr	Dvořáček	11.8.2010 16:41:29	02	02	30.6.2010	0	Odhod	01	01642BEE120000F3	Ne		0
Zdeněk	Sotolář	11.8.2010 16:54:23	01	01	30.6.2010	0	Odhod	01	019990EC120000F3	Ne		0
Petr	Dvořáček	12.8.2010 6:55:37	02	02	30.6.2010	0	Odhod	01	01642BEE120000F3	Ne		0
Petr	Dvořáček	12.8.2010 15:04:04	02	02	30.6.2010	0	Odhod	01	01642BEE120000F3	Ne		0
Petr	Dvořáček	12.8.2010 15:13:45	02	02	30.6.2010	0	Odhod	01	01642BEE120000F3	Ne		0
Martin	Dudáček	12.8.2010 17:33:09	04	04	30.6.2010	0	Odhod	01	018985EF12000063	Ne		0
Zdeněk	Sotolář	12.8.2010 17:33:27	01	01	30.6.2010	0	Odhod	01	019990EC120000F3	Ne		0
Martin	Dudáček	13.8.2010 15:01:32	04	04	30.6.2010	0	Odhod	01	018985EF12000063	Ne		0
Petr	Dvořáček	13.8.2010 15:08:33	02	02	30.6.2010	0	Odhod	01	01642BEE120000F3	Ne		0
Zdeněk	Sotolář	13.8.2010 16:18:31	01	01	30.6.2010	0	Odhod	01	019990EC120000F3	Ne		0
Petr	Dvořáček	16.8.2010 14:15:20	02	02	30.6.2010	0	Odhod	01	01642BEE120000F3	Ne		0

Obrázek 2–4 - Software Docházka MINI dodávaný firmou ASC line [2]

V nabídce jsou pro ADS4 moduly, jako například INTRAWEB, sloužící k prohlížení docházky a monitoringu přítomných osob pomocí webového prohlížeče. Další nabízený modul VISITOR je určený k evidenci návštěv pohybujících se po objektu. Cena licence softwaru ADS4 je bez dodatečných modulů 17 800Kč pro 100 zaměstnanců.

## 2.3 Miaxis biometrics – Attendance management

Jako další příklad byl zvolen zahraniční výrobce a dodavatel docházkových a přístupových systému, společnost Miaxis biometrics Co.,Ltd. Tato firma vyrábí a dodává též samostatné moduly optických čteček otisků prstu, které lze nalézt v produktech jiných firem a také v terminálu, jehož vývojem se zabývá tato práce. V následujících podkapitolách věnujících se produktům Miaxis biometrics jsou použity materiály převzaté z [3].

### 2.3.1 Terminály

Miaxis má v nabídce několik typů terminálů, pro srovnání byl zvolen jeden z větších modelů v nabídce, vybavený krom optického senzoru otisku prstu také klávesnicí a grafickým LCD displejem.

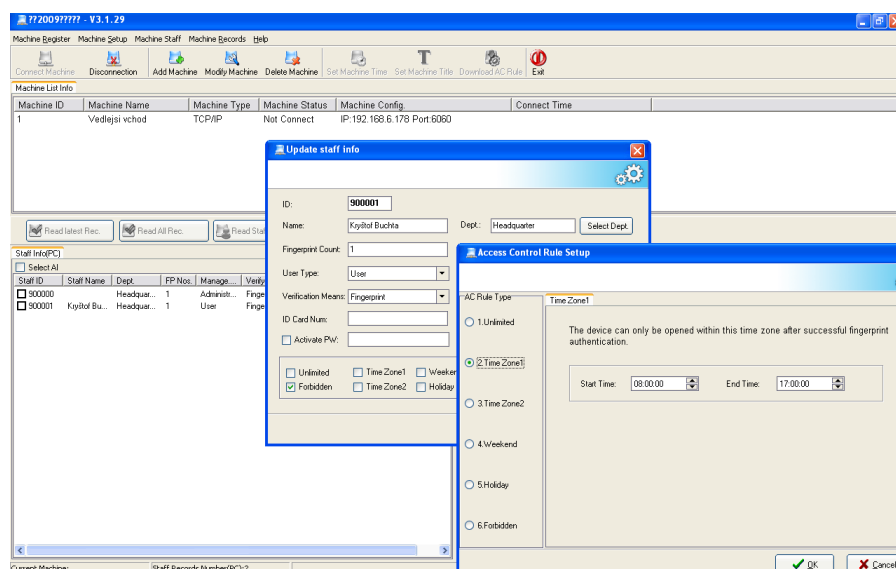


Obrázek 2-5 - Terminál Miaxis MX-600 [3]

- Terminál je nabízen ve 3 provedeních komunikačního rozhraní – USB, RS232, TCP/IP
- Paměť pro 3000 zaměstnanců a 130 000 záznamů
- Rozhraní wiegand-34
- Použitý optický senzor – Miaxis SM-621
- Napájení 12V, pracovní odběr 450mA, klidový 100mA
- Cena 160USD (při kurzu ze dne 17.5.2013 se jedná o cca 3225 Kč)

### 2.3.2 Software

Na přiloženém CD se nachází instalační soubor programu *Miaxis 2009 attendance management system*. Ten je koncipován jako aplikace pro operační systém Windows. Umožňuje přidávat a mazat uživatele z terminálů nebo číst záznamy o jejich docházce.



Obrázek 2-6 - Software dodávaný firmou Miaxis k terminálům [3]

## 3 KONCEPCE SYSTÉMU

Samotnému vývoji každého zařízení by měla předcházet po analýze trhu důkladná fáze příprav. V ní je nutno jasně formulovat požadavky na daný produkt, které vycházejí především z aktuální situace na trhu a odvíjí se od plánovaného nasazení výrobku.

### 3.1 Typické nasazení

Jak již bylo v úvodní kapitole zmíněno, předpokládané nasazení vyvíjeného docházkového systému je v menších společnostech, typických počtem uživatelů v řádu jednotek či desítek. Vzhledem k tomu, že mzdové výkazy či jiné operace s nasbíranými záznamy se provádí ve většině firem zpětně za kalendářní měsíc, je třeba dimenzovat paměť terminálu pro uchování docházkových dat alespoň za 2 měsíce. Pro účely hrubého odhadu uvažujme na jednoho uživatele 6 zaznamenaných událostí za pracovní den (např. příchod, odchod, oběd, kuřácká pauza). Velmi přibližný odhad objemu dat v systému by pak mohl vypadat následovně:

$$(6 \text{ záznamů uživatel / den}) \times (100 \text{ uživatelů}) \times (2 \text{ měsíce} \approx 62 \text{ dní}) = 37\,200 \text{ záznamů}$$

V reálu lze předpokládat objem dat spíše nižší, protože nejsou všechny dny pracovní a ne všichni zaměstnanci přerušují dvakrát denně pracovní dobu. Terminál docházkového systému bývá zpravidla instalován poblíž hlavního vchodu do podniku. Nejčastější způsob instalace je pak uchycení na stěnu.

### 3.2 Požadavky

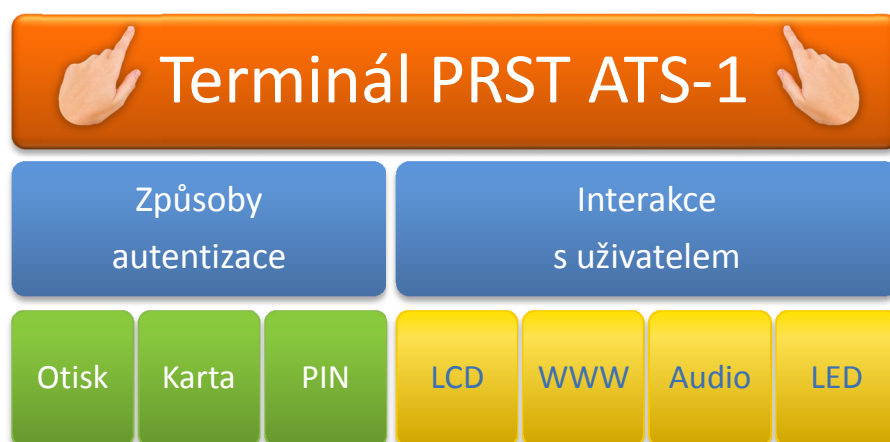
Na základě průzkumu trhu v oblasti docházkových systémů byl zvolen následující mix obecných priorit, na něž je během vývoje kladen důraz:

- **Snadná rozšiřitelnost** – Cílem je dosažení stavu kdy se nepřizpůsobuje aplikace zařízení, ale zařízení aplikaci. Hlavními nástroji k dosažení tohoto cíle jsou univerzálnost a modularita. U terminálu pro docházkový systém lze tyto požadavky řešit na 2 úrovních – HW a SW. U hardwaru se bude jednat o širokou nabídku rozhraní pro komunikaci s okolím a také o možnost rozšíření funkcionality pomocí zásuvného modulu (karty). Co se týče softwaru, je v tomto ohledu velice výhodné využití standardizovaného operačního systému. Ten vytvoří patřičné prostředí pro běh programových modulů potřebných v dané aplikaci. Dále bude požadován vzdálený přístup, umožňující provádět změny v softwarové výbavě bez nutnosti fyzické přítomnosti technika.

- **Příjemné uživatelské rozhraní** – Je již v současné době uživateli považováno za samozřejmost a vyžadováno u téměř všech zařízení. Děje se tak především díky masovému rozšíření zařízení spotřební elektroniky jako jsou například tzv. chytré telefony či tablety. Co se týká terminálu docházkového systému s webovým rozhraním, je rozhodně výhodné (ne-li téměř nutné) použít tam, kde dochází ke kontaktu s uživatelem soudobých technologií. Požadován je tak v každém případě barevný dotykový displej či webserver s dynamickým webovým rozhraním.
- **Nízké náklady na pořízení i provoz** – Nízké pořizovací ceny zařízení lze dosáhnout více cestami. Jednou z nich je minimalizace ceny či množství použitých komponent. To je výhodné zejména u velkosériové výroby, kde jsou fixní náklady na vývoj rozděleny do velkého množství kusů a převážnou část ceny tak tvoří součástky. Druhý způsob pak spočívá v minimalizaci nákladů na vývoj, které naopak u kusové či malosériové výroby můžou tvořit i naprostou většinu ceny zařízení. Při vývoji terminálu pro docházkový systém bude pozornost věnována oběma způsobům minimalizace ceny.

Na základě výše uvedených obecných požadavků a typického nasazení lze přikročit ke konkrétní formulaci požadované funkcionality. Jako primární identifikační faktor bude vzhledem k vyššímu stupni zabezpečení oproti jiným způsobům využit otisk prstu. Sekundární alternativu bude tvořit čtečka RFID karet a čtyřmístný PIN (např. pro prostředí se zvýšeným znečištěním). Z důvodu snazšího zneužití se však jedná o méně bezpečné způsoby identifikace.

Jak vyplývá ze zadání (a názvu) této práce, bude jeden kanál komunikace s uživatelem realizován prostřednictvím webového rozhraní. Sloužit má především ke správě systému a čtení nasbíraných záznamů od docházce. Z obecných požadavků, formulovaných výše, vyplývá využití barevného dotykového displeje pro hlavní kontakt s osobou, která se fyzicky nachází u terminálu či jej zrovna obsluhuje. LCD zobrazovač bude doplněn přídatnými kanály, tvořenými zvukovým výstupem a dvojbarevnou LED.



Obrázek 3–1: Způsoby autentizace a kanály komunikace s uživatelem

## 4 HARDWARE TERMINÁLU

Na základě funkcionality nabízené obdobnými systémy dostupnými na trhu (viz *kapitola 2 Existující řešení*) a na základě stanovených priorit (viz *kapitola 3 Koncepce systému*) byla po dohodě s vedoucím této práce stanovena požadovaná hardwarová výbava terminálu:

- Výpočetní systém s dostatečným výkonem a pamětí
- Barevný dotykový displej
- Čtečka otisků prstu
- Snímač bezkontaktních karet
- Rozhraní ethernet
- Klávesnice
- Pohybový senzor
- Audiovýstup
- Binární I/O rozhraní (galvanicky oddělené)
- Sériové rozhraní pro externí využití
- Možnost rozšíření pomocí zásuvného modulu
- Záložní zdroj energie

V následujících podkapitolách budou jednotlivé hardwarové celky podrobněji rozebrány.

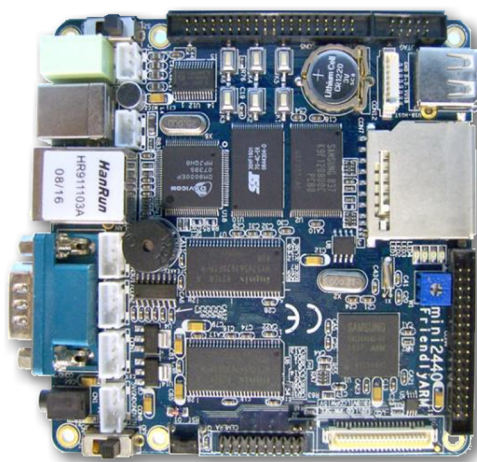
### 4.1 Kit MINI2440

Klíčovou součástí a „srdcem“ terminálu je bezpochyby mikroprocesorová část. Při její volbě vycházejme z činností, které bude v aplikaci zajišťovat. Zohledněme tedy požadavky popsané v *kapitole 3.2*, mezi něž patří obsluha barevného dotykového displeje, audio výstupu, poskytování dat formou dynamických webových stránek či vzdálená správa. Jako velice výhodná se zde jeví volba platformy umožňující provoz operačního systému (který nemusí pro dané účely nutně pracovat v reálném čase). Do aplikace popisované v této práci byl vzhledem k množství dostupného softwaru, open – source formě či podpoře mnoha HW platforem vybrán OS Linux.

Na základě předchozích pozitivních zkušeností autora byla zvolena platforma MINI2440 od FriendlyARM [5]. Jedná se o jednodeskový počítač s OS Linux, jehož výhody a vlastnosti byly podrobně popsány v autorově předchozí práci [4]. Nyní tedy bude uveden pouze stručný soupis nejdůležitějších technických parametrů:

- Procesor s jádrem ARM9 Samsung 3SC2440 taktovaný na 400 MHz
- Paměť RAM 64 MB

- Programovou paměť flash typu NAND o kapacitě 256 MB
- Ethernetový řadič 10/100Mb
- 3 sériové linky, z nichž jsou dvě v úrovních TTL a jedna RS232
- Zvukový kodek AC97 s výstupem na 3.5 mm jack
- Obvod reálného času zálohovaný baterií CR1216
- 1x USB-device, 1x USB-host, slot na SD kartu
- Jednoduché periferie jako tlačítka, LED, bzučák, sériová eeprom
- 3.5“ dotykový barevný LCD panel (rezistivní princip) s rozlišením 320x240 px



Obrázek 4–1: Deska MINI2440 od FriendlyARM (bez LCD) [5]

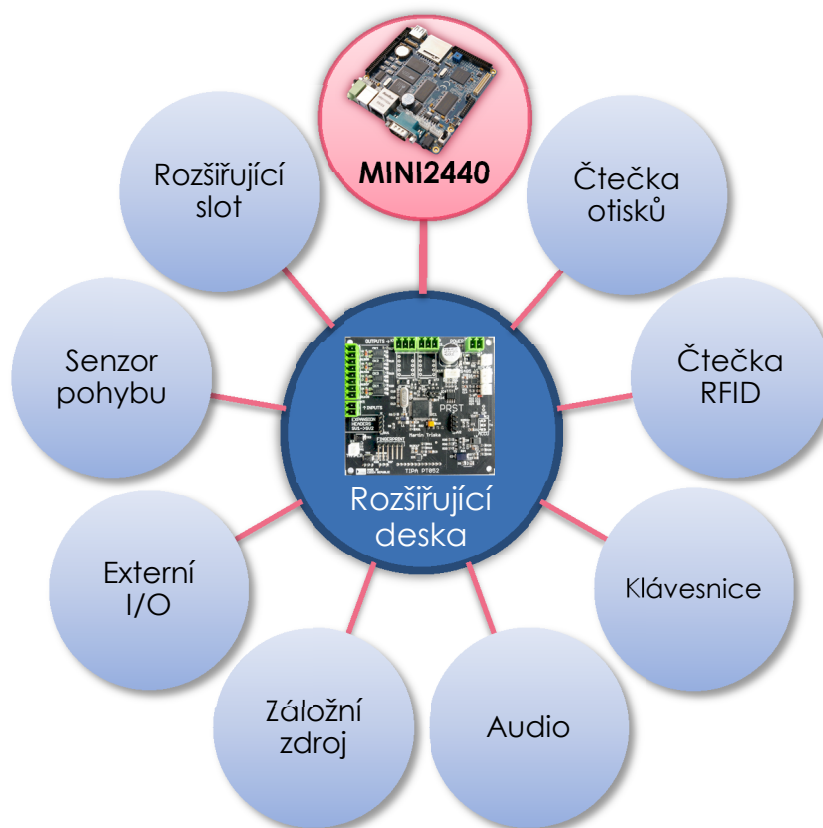
Tento vývojový kit umožňuje provoz různých OS, mezi nimiž nechybí výše zmiňovaný Linux. Nevýhodou MINI2440 je relativně zastaralý CPU, který má ve srovnání s moderními jádry ARM Cortex nižší výkon při vyšší spotřebě energie.

## 4.2 Rozšiřující deska

Jelikož zvolený vývojový kit MINI2440 neobsahuje dostatečné množství komunikačních rozhraní a dalších periférií potřebných v této aplikaci, je nezbytné navrhnout a vyrobit rozšiřující modul. Jako rozhraní po komunikaci s MINI2440 byla zvolena sériová linka typu UART. Výhodou tohoto řešení je možnost využít rozšiřující desku i s jiným nadřazeným systémem, případně v jiné aplikaci. Potřebné je pouze rozhraní UART, jímž bývá v některé z jeho podob většina embedded platforem vybavena, případně ho lze emulovat, např. pomocí USB.

Krom prostého rozšíření vstupů/výstupů umožní rozšiřující deska také provádět některé rutinní operace, které tak nebudou zbytečně zatěžovat nadřazený systém. Jedná se například o funkce spojené s uživatelským rozhraním jako je obsluha klávesnice, čtečky otisků či karet. Dále mezi úlohy realizované rozšiřující deskou patří správa

záložního napájení z akumulátoru a možnost osazení rozšiřující karty (např. pro bezdrátovou konektivitu).

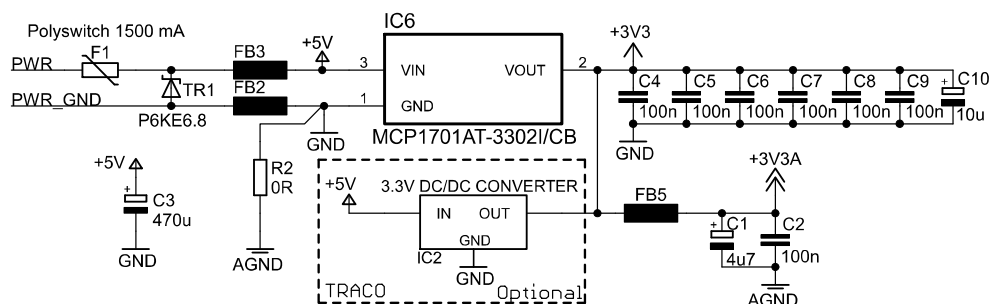


Obrázek 4-2- Úlohy rozšiřující desky v terminálu

V následujících podkapitolách budou popsány jednotlivé funkční celky rozšiřující desky a zdůvodněna jejich volba.

### 4.2.1 Napájení

Drtivá většina současných mikrokontrolérů (dále MCU) i dalších integrovaných obvodů je schopna provozu při napájecím napětí 3,3 V, avšak výkonově orientované obvody či celé konstrukční moduly pracují často také s napětím 5 V. Proto musí být na rozšiřující desce k dispozici obě uvedená napětí. Stabilizovaných 5 V bylo zvoleno jako hlavní napájecí napětí terminálu. Výhodou je dobrá dostupnost spínaných zdrojů pro toto napětí, a oproti napětí vyššímu také možnost vynechání jednoho snižujícího stupně. Nevýhody pak vyplývají z vyššího odebíraného proudu a následným vyšším ztrátám na napájecích přívodech. V případě napájecí trasy s větší impedancí (např. delší či slabé vodiče, typicky PoE) by bylo vhodné provést konverzi z vyššího napětí na 5 V co nejblíže terminálu. Při návrhu mechanického uspořádání je na tuto možnost pamatováno vyčleněním prostoru na konverzní DC/DC modul.



Obrázek 4–3: Napájení rozšiřující desky

Co se týče konverze napětí 5 V na 3,3 V, poskytuje deska výběr ze dvou možností. Pro variantu s velmi nízkým proudovým odběrem z 3,3 V větve (žádný přídavný modul, využívání nízkospotřebných režimů MCU...) je výhodné osazení low-drop lineárního stabilizátoru. Zvolen byl typ MCP1701AT (IC6) od výrobce Microchip pro svůj velmi nízký klidový proud spotřebovaný samotným stabilizátorem — 2 $\mu$ A [6]. Pro minimální proudové odběry je tedy v tomto případě účinnost vyšší, než při použití výrazně dražšího DC/DC měniče, který se vyznačuje klidovým odběrem v řádu jednotek mA [7]. Pokud jsou proudové nároky aplikace vyšší, je již snižující DC/DC měnič nutností. Vybrán byl typ TSR 1-2433 (IC2) od výrobce Traco Power s maximálním proudem 1 A [7].

Napájecí napětí 5 V je přivedeno na svorkovnici K1, odkud pokračuje přes vratnou pojistku polyswitch (jejíž hodnota závisí na konkrétní HW konfiguraci, v popisované verzi byla dimenzována na 1500 mA), transil a odrušovací feritové perly FB2 a FB3 na stabilizátor IC6 (či volitelně IC2). 5 V větev filtruje kondenzátor C3, na jehož místě je osazen kvalitní nízkoimpedanční (*LowESR*) typ. Ze stabilizovaného napětí 3,3 V je dále pomocí samostatného filtru odvozeno napájení AD převodníku, které ve schématu označuje symbol +3V3A. Analogová zem GNDA má z důvodu eliminace zemních smyček s hlavní GND pouze jeden styčný bod.

## 4.2.2 Mikrokontrolér

Řídicí prvek rozšiřující I/O desky tvoří mikrokontrolér STM23F100RCT6B od výrobce STM. Jedná se o 32bitový MCU s jádrem ARM Cortex M3. Zvolen byl především pro svoji širokou škálu komunikačních rozhraní a relativně nízkou cenu. Tento model spadá to takzvané *Value-line*, tj. do nejnižší řady vybavené tímto jádrem, která je u výrobce k dispozici. Výpočetní výkon i nabídka periférií však pro potřeby rozšiřující I/O desky naprosto dostačují, a to i s ohledem na prostor pro další rozšiřování. Hlavní technické parametry [8]:

- Maximální frekvence 24 MHz, výpočetní výkon 1.25 DMIPS/MHz
- Paměť flash 256 kB, RAM 24 kB
- Komunikační rozhraní:
  - 3xUSART



- 2xUART
- 2xI2C
- 3xSPI
- 16-kanálový A/D převodník, 2-kanálový D/A převodník
- 51 GPIO linek
- Pouzdro LQFP64

Zapojení plně respektuje doporučení uvedená v aplikační poznámce [9]. Napájecí napětí 3,3 V je v blízkosti obvodu filtrováno tantalovým kondenzátorem s kapacitou 10  $\mu$ F. Následně každý pár napájecích svorek blokuje keramický kondenzátor o hodnotě 100 nF. Separátní přívod napájení AD převodníku využívá zdroj napětí odvozený z hlavního napájecího, filtrovaného navíc tantalovým kondenzátorem o kapacitě 4,7  $\mu$ F a blokovaného stejně jako ostatní napájecí vstupy MCU (viz *Obrázek 4–3*). Navíc je předřazena feritová perla, jejíž přesná hodnota indukčnosti není na této pozici kritická (zvoleno bylo 100  $\mu$ H).

Zdroj hodinového kmitočtu tvoří krystalový oscilátor o frekvenci 8MHz. Zde je nutno provést často opomíjený výpočet pasivních součástí dle doporučení výrobce [10]. Byl zvolen běžně dostupný krystal typu HC-49S (prodává např. GMe) o kmitočtu 8 MHz. Tento krystal má následující parametry [6]:  $C_L = 20$  pF;  $C_0 = 7$  pF;  $ERS = 50$   $\Omega$ . Pro výpočet je také třeba znát transkonduktanci oscilátoru v MCU, dle katalogového listu [11]  $g_m = 25$  mA/V. Výpočet kritické hodnoty  $g_m$  dané parametry krystalu [10]:

$$g_{m_{crit}} = 4 \times ESR \times (2\pi f)^2 \times (C_0 + C_L)^2 \quad (4.2.1)$$

$$g_{m_{crit}} = 4 \times 50 \times (2\pi \cdot 8 \cdot 10^6)^2 \times (7 \cdot 10^{-12} + 20 \cdot 10^{-12})^2 = 0,37 \text{ mA/V} \quad (4.2.2)$$

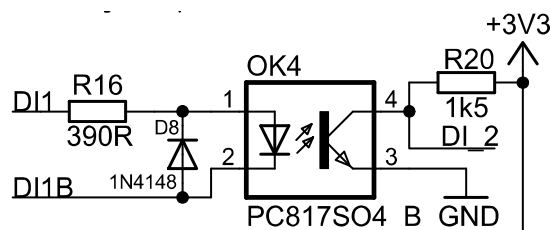
Vypočtená hodnota potvrzuje splnění základní podmínky oscilace  $g_m \gg g_{m_{crit}}$ . Externí zatěžovací kondenzátory jsou dány paralelní kombinací, jejíž velikost se rovná zatěžovací kapacitě krystalu  $C_L$  snížené o parazitní kapacitu DPS a vývodů pouzdra čipu. Ta byla odhadnuta na 5pF. Za předpokladu  $C_{L1} = C_{L2}$  lze psát zjednodušený vztah:

$$C_{L1} = C_{L2} = 2 \cdot (C_L - C_s) = 2 \cdot (20 \text{ pF} - 5 \text{ pF}) = 30 \text{ pF} \quad (4.2.3)$$

Pin NRST sloužící k vyvolání restartu mikrokontroléru není v této aplikaci využit, protože MCU obsahuje funkci POR (*power on reset*) a PDR (*power down reset*). NRST je pouze dle doporučení výrobce [9] připojen přes kapacitu 100 nF na GND jako ochrana před možným náhodným restartem. Obdobně nejsou využity piny VBAT (sloužící pro připojení baterie zálohující část RAM) a BOOT1 (aktivace režimu bootloaderu – ten je aktivován softwarovou cestou).

### 4.2.3 Binární vstupy a výstupy

Osazeny jsou 4 binární vstupy, galvanicky oddělené pomocí optočlenů typu PC817B. Každý vstup obsahuje ochrannou diodu, zamezující zničení optronu při přivedení signálu opačné polaritě a rezistor určující proud vstupní stranou optočlenu. Z jeho hodnoty se odvíjí rozsah vstupního napětí. Maximální proud tekoucí do vstupu optočlenu je 50 mA, volbu proudu pracovního však ovlivňuje několik faktorů.



Obrázek 4–4: Galvanicky oddělený vstup

Pomineme-li snahu o minimalizaci výkonové ztráty, hraje zde roli stárnutí a degradace optočlenu, především pak jeho vstupní části (tímto jevem se zabývá např. [12]). Hlavním projevem je zhoršení proudového transformačního poměru CTR (*current transfer ratio*). S touto skutečností je třeba v návrhu počítat a nevolit vstupní proud optočlenem na minimální možné hranici. Avšak příliš vysoký proud blížící se proudu maximálnímu proces stárnutí urychluje, a proto byl pro účely této práce zvolen jako kompromis proud  $I_f = 10 \text{ mA}$ . Při předpokládaném pevném vstupním napětí 5 V a napětí na LED v propustném směru 1,2 V [13] lze určit hodnotu předřadného rezistoru R16 (a odpovídajících rezistorů u zbývajících vstupů) následovně:

$$R_p = \frac{U - U_F}{I_F} = \frac{5 \text{ V} - 1,2 \text{ V}}{10^{-2} \text{ A}} = 380 \Omega \quad (2.2.4)$$

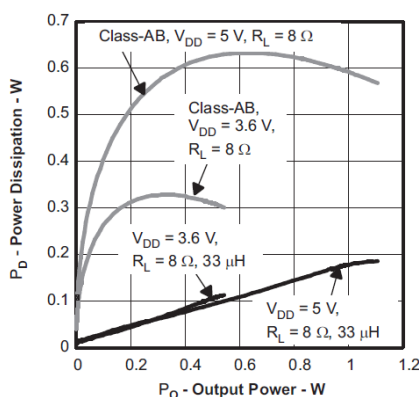
Dále je třeba u vstupní části optočlenu zvolit proud tekoucí fototranzistorem. Protože použitý MCU umožňuje provozovat GPIO piny ve vstupním režimu s vnitřními pull-up (o hodnotě 40 kΩ, viz [13]), není osazení externích rezistorů naprosto nezbytné. V tomto projektu však použity byly. Jejich hodnota 1,5 kΩ odpovídá proudu fototranzistorem  $I_C = 2 \text{ mA}$ . Zde není volba přesných hodnot kritická, postačí, aby byl tranzistor pro danou kombinaci  $I_C$  a  $I_F$  plně otevřen, což lze určit z průběhu [13] Fig. 9.

Rozšiřující deska obsahuje dále 2 binární přepínací výstupy, realizované miniaturními relé typu P-5 výrobce Forward-Relays. Jejich maximální proudové zatížení je 2 A / 30 VDC nebo 0,5 A / 125 VAC [14], což plně dostačuje například pro magnetický dveřní zámek.

## 4.2.4 Audio zesilovač

Vývojový kit MINI2440 disponuje zvukovým výstupem, vhodným například pro připojení sluchátek. K zajištění hlasitého audiovýstupu pomocí reproduktoru je však nutno linkovou úroveň výstupu MINI2440 výkonově zesílit. Vzhledem k dostatečné nabídce integrovaných obvodů nemá pro tuto aplikaci význam přemýšlet o řešení z diskrétních součástek. Limitujícím faktorem je zde napájecí napětí, které je 5V.

V současné době se v domácí i profesionální sféře začínají stále více prosazovat zesilovače pracující v pulzním (PWM) režimu, označované zpravidla jako třída „D“. Výhodné jsou především pro svoji vysokou účinnost v porovnání s „tradičním“ lineárním řešením (typicky třídou AB).



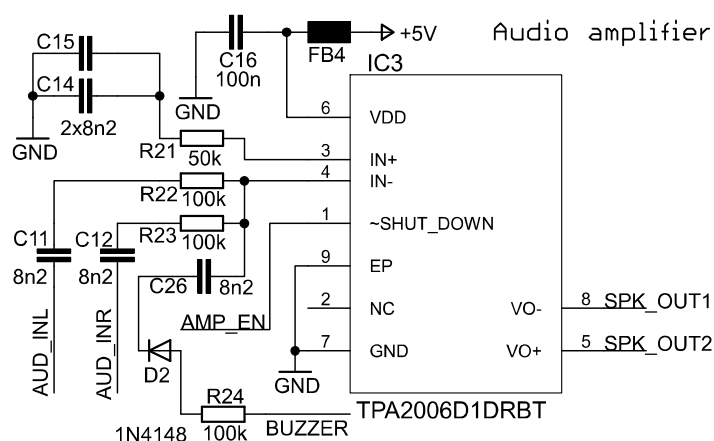
Tabulka 1 - Srovnání ztrátového výkonu zesilovačů ve třídě AB a ve třídě D (TPA2006) [15]

Zesilovač ve třídě D byl zvolen i pro použití v terminálu docházkového systému PRST. Jedná se o obvod TPA2006 [15] od výrobce Texas Instrumens, který je optimalizován pro bateriově napájená zařízení, jako jsou mobilní telefony a PDA. Vyznačuje se možností tzv. *filterless* zapojení, kdy pro krátkou délku kabelu k reproduktoru (do 10cm) není třeba osazovat výstupní filtr, potlačující pracovní frekvenci obvodu a její vyzařování do okolí (včetně vyšších harmonických).

Výstupní výkon	<b>1,19W</b> ( $R_L=8\Omega$ , $V_{DD}=5V$ , THD=1%, $f=1kHz$ )
Napájecí napětí	2,5 – 5,5V
Proudový odběr	3,4mA/klid. Proud 0,5 $\mu$ A/shutdown režim
Účinnost	86% ( $P = 1W$ , $R_L = 8\Omega$ , $V_{DD} = 5V$ )
Pracovní frekvence	250kHz
Pouzdro	QFN20
Cena	0,8 € (20. duben 2013, Mouser, 1 ks)

Tabulka 2 - Základní parametry obvodu TPA2006 [15]

Zesilovač TPA2006 obsahuje symetrický (diferenciální) vstup. Ten je třeba pro připojení ke kitu MINI2440 upravit – převést na nesymetrický, označovaný jako *single-ended*. To se provede připojením neinvertujícího vstupního pinu na nulový potenciál. Signály levého a pravého kanálu jsou sečteny a přivedeny na vstup přes RC členy. Zesílení udává velikost rezistoru, který spolu s vazebním kondenzátorem tvoří hornopropustný RC filtr záměrně potlačující nízké kmitočty, jež použitý reproduktor není schopen vyzářit. Nejprve je třeba zvolit zesílení zapojení, přičemž příliš nízká hodnota neumožní dostatečné vybudnění a naopak hodnota vysoká bude znamenat horší odstup rušivých signálů a větší pravděpodobnost přebuzení.



Obrázek 4-5- Zapojení zesilovače TPA2006D1

Kit MINI2440 obsahuje zvukový kodek Philips UDA1341TS [14], napájený 3,3 V. Obvod při napájení 3 V poskytuje výstupní napětí 0,9 V<sub>RMS</sub>, které je lineárně závislé na napětí napájecím. S jistou chybou lze tak výstupní napětí považovat za 1 V. Pro jmenovitý výstupní výkon 1,19 W na zátěži lze vypočítat efektivní hodnotu napětí na zátěži (reproduktoru):

$$P = U \cdot I = \frac{U^2}{R} \Rightarrow U = \sqrt{P \cdot R} = \sqrt{1,19W \cdot 8\Omega} \cong 3V \quad (2.2.5)$$

Ze známého vstupního i výstupního napětí je pak vypočten požadovaný napěťový zisk:

$$A_U = \frac{U_2}{U_1} = \frac{3V}{1V} = 3 \quad (2.2.6)$$

Vstupní rezistor pro zisk, umožňující plné vybudnění pak bude:

$$A_U = \frac{2 \cdot 150\Omega}{R_I} \Rightarrow R_I = \frac{300k\Omega}{A_U} = \frac{300k\Omega}{3} = 100k\Omega \quad (2.2.7)$$

V případě že postačí nižší dosažitelná úroveň hlasitosti, lze samozřejmě zisk snížit osazením většího rezistoru.

Nyní bude vypočtena mezní frekvence vstupního filtru typu horní propust. Jelikož výstup bude využíván především pro hlasové pokyny a informace, postačí spodní frekvenci nastavit na hodnotu 200Hz.

$$C_I = \frac{1}{2 \cdot \pi \cdot R_I \cdot f_c} = \frac{1}{2 \cdot \pi \cdot 100k\Omega \cdot 200Hz} \cong 8nF \quad (2.2.8)$$

Protože hodnota 8nF není běžně dostupná, bude osazena hodnota 8,2nF v provedení keramickém (s dielektrikem NP0) či svitkovém. Analogickým způsobem byly vypočteny hodnoty pasivních součástek pro třetí vstup BUZZER (PWM signál z procesoru). Ten je navíc opatřen ochrannou diodou.

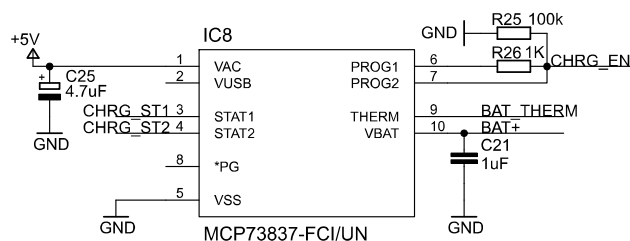
## 4.2.5 Nabíječ Li-Ion akumulátoru

U akumulátorů typu Li-Ion a Li-Pol je důsledkem jejich velkého rozšíření ve výrobcích spotřební elektroniky poměrně široká nabídka podpůrných obvodů, jako jsou měniče napětí, čítače kapacity a v neposlední řadě také nabíjecí IO. Jako nabíjecí obvod byl zvolen typ MCP73837 od výrobce Microchip [16]. Jedná se o plně integrovaný nabíječ samostatných li-ion a li-pol článků, vyžadující pro svou práci minimum externích součástek. Jeho základní parametry udává následující tabulka:

Výstupní napětí	4.2V ± 0.5%
Vstupní napětí	4.5 – 6V
Nabíjecí proud	15-1000mA (nastavitelné omezení)
Princip nabíjení	Konstantní proud, poté konstantní napětí
Zpětný proud z baterie	0.1μA (Standby režim)
Pouzdro	MSOP-10
Cena	1,03 € (20. duben 2013, Mouser, 1 ks)

Tabulka 3 - Základní parametry obvodu MCP73837 [16]

Obvod nabíjí akumulátor vícefázovým procesem podle pevně daného a relativně sofistikovaného algoritmu, jehož stavový diagram lze nalézt v katalogovém listu [16]. K dostání je ve více variantách, jenž se liší parametry jako výstupní napětí či mez proudu, působící ukončení nabíjecího cyklu.



Obrázek 4–6: Zapojení nabíjecího obvodu MCP73837

Třístavové vstupy PROG1 a PROG2 definují nabíjecí proud v režimech nabíjení ze sítě nebo USB. Těmito vstupy lze také uvést obvod do vypnutého (*Shutdown*) či pohotovostního (*Standby*) stavu. Dále MCP73837 obsahuje trojici signálů, použitelnou k informování nadřazeného systému o aktuálním režimu obvodu (*shutdown*, *standby*, *charging*, *charging complete*).

Proud v režimu nabíjení konstantním proudem je spojen s velikostí rezistoru R47 vztahem [16]:

$$I_{REG} = \frac{1V}{R/1000} = \frac{1V}{1000\Omega/1000} = 1A \quad (2.2.9)$$

Pro nejvyšší možný nabíjecí proud 1000mA je tedy na pozici R47 třeba osadit rezistor 1kΩ. Jeho konkrétní hodnota však vychází z použitého akumulátoru, kdy je zpravidla nabíjecí proud doporučen výrobcem.

## 4.2.6 DC/DC měnič záložního zdroje

Zvolen byl specializovaný obvod TPS61252 výrobce Texas Instrumens [17]. Jeho typickou aplikací je konverze proměnného napětí jednoho Li-Ion článku na napětí 5V, což je napájecí napětí terminálu.

Vstupní napětí	2,3 V – 6 V
Výstupní napětí	3 V – 6,5 V
Odebíraný proud	1500 mA (nastavitelné omezení 100 – 1500 mA)
Účinnost	90 % ( $V_{IN} = 3,6 \text{ V}$ ; $V_{OUT} = 5 \text{ V}$ ; $I_{out} = 700 \text{ mA}$ )
Pracovní frekvence	3.25 MHz
Pouzdro	QFN-8
Cena	2,75 € (20. duben 2013, Mouser, 1 ks)

Tabulka 4 - Základní parametry obvodu TPS61252 [17]

Zapojení tohoto obvodu vychází ze zapojení katalogového [17]. Výstupní napětí udává dělič, složeným z rezistorů R27 a R28. Interní referenční napětí  $V_{FB}$  je 1,2 V. Rezistory bylo nutno vybrat z dostupných hodnot.

$$V_{OUT} = V_{FB} \left( 1 + \frac{R1}{R2} \right) = 1.2V \left( 1 + \frac{750k\Omega}{240k\Omega} \right) = 4.95V \quad (2.2.10)$$

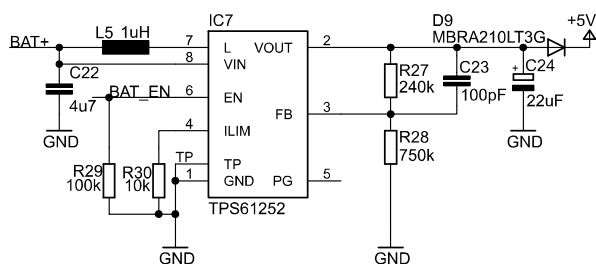
Jelikož v době návrhu zapojení nebyly přesně známy celkové proudové nároky systému, bylo odhadnuto omezení proudu odebíraného z akumulátoru na hodnotu 1 A. Maximální výstupní proud je v tomto případě dán následovně:

$$I_{out(max)} \approx I_{LIM} \left( \frac{V_{IN} \cdot \eta}{V_{OUT}} \right) = 1A \left( \frac{3.7V \cdot 0.9}{4.95V} \right) = 670mA \quad (2.2.11)$$

Maximální odebíraný proud lze nastavit neosazením rezistoru R44 a přímým spojením vývodu ILIM s výstupem TPS61252. V případě této úpravy je pak výstupní proud (při maximálním odebíraném proudu z akumulátoru 1500 mA):

$$I_{out(max)} \approx I_{LIM} \left( \frac{V_{IN} \cdot \eta}{V_{OUT}} \right) = 1500mA \left( \frac{3.7V \cdot 0.9}{4.95V} \right) = 1000mA \quad (2.2.12)$$

Tyto výpočty jsou spíše orientační a platí pro konkrétní pracovní bod – jmenovité napětí akumulátoru. Plně nabitý bude mít napětí jiné než vybitý těsně pod odpínací napěťovou mez.



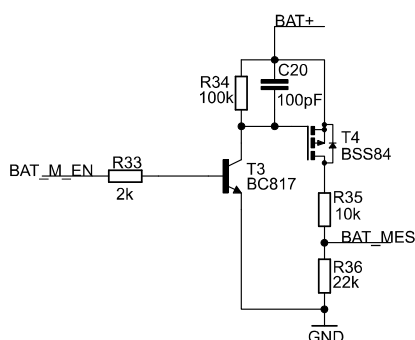
Obrázek 4-7 - Zapojení obvodu TPS61252

Rezistor R29 tvoří tzv. pull-down na vstupu EN, zabráňující samovolnému spuštění měniče. Speciální nízko-úbytková schottky dioda D9 funguje jako ochrana v případě, kdy by došlo k současnému sepnutí hlavního i záložního zdroje. Vytvořený úbytek v propustném směru  $V_F = 0,325 V$  (při proudu 1 A a teplotě 25° C, viz [18]) snižuje napájecí napětí systému, čehož je využito pro identifikaci znovupřipojení síťového napájecího napětí po jeho výpadku.

## 4.2.7 Měření napětí

Aby bylo možno aktivovat záložní zdroj včas, aniž by došlo k přerušení dodávky napájení pro komponenty terminálu, je nezbytné neustále monitorovat hlavní napájecí napětí. To zajišťuje A/D převodník MCU, který pro tento účel poskytuje periferii zvanou *analog watchdog*. Ta periodicky porovnává napětí přivedené na zvolený vstup A/D převodníku s konstantními mezemi, uloženými v příslušných registrech, a pokud se naměřené napětí dostane mimo nastavené rozmezí, vyvolá se přerušení. Jelikož je referenční napětí AD převodníku odvozeno z napájecího napětí MCU 3,3 V, je nutné použít při měření napájecího napětí terminálu odporový dělič. Zvolen byl dělicí poměr 1:1 z důvodu ponechání určité rezervy pro případ přepětí. S tímto dělicím poměrem činí rezerva cca 25% rozsahu A/D převodníku ( $2,5\text{ V} \approx 75\% \text{ z } 3,3\text{ V}$ ).

V případě, kdy délka výpadku napájení přesáhne čas, po který je schopen záložní zdroj dodávat energii, musí terminál ukončit svoji činnost kontrolovaným způsobem. Aby bylo možné signalizovat blížící se vyčerpání akumulátoru, je třeba měřit jeho napětí (lepší řešení by bylo sledování vyčerpaného a zbývajících náboje, což umožňují specializované obvody označované jako tzv. *fuel gauge*, například bq27541 od Texas Instruments). Jak již bylo zmíněno, zvolený MCU využívá jako referenci AD převodníku napájecí napětí 3,3 V, tudíž napětí akumulátoru je třeba pro jeho změření rovněž snížit. Prosté připojení odporového děliče paralelně k akumulátoru není vhodným řešením z důvodu jeho trvalého vybíjení příčným proudem děliče. Částečným řešením je osazení velmi vysokých hodnot rezistorů, kdy však roste náchylnost na okolní rušení a klesá přesnost měření. Pro tento projekt bylo využito zapojení se spínaným napěťovým děličem (obdobu [19]):



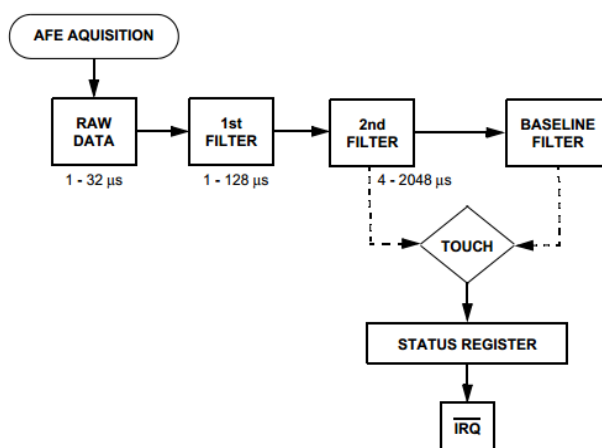
Obrázek 4–8: Spínaný napěťový dělič

Dané zapojení eliminuje nutnost osazovat velmi vysoké hodnoty rezistorů do měřicího odporového děliče, který je připojen pouze v okamžiku měření. Dělicí poměr byl zvolen přibližně 2:1 (odpor otevřeného kanálu unipolárního tranzistoru lze při těchto hodnotách rezistoru zanedbat).



## 4.2.8 Řadič kapacitní klávesnice

Z principu lze samozřejmě realizovat kapacitní klávesnici bez speciálního obvodu, pouze pomocí GPIO/ADC pinů MCU. V takovémto případě je však pro zajištění vysoké a především dlouhodobě stabilní spolehlivosti čtení nutno využít poměrně rozsáhlých a na procesorový čas náročných algoritmů. Vzhledem k úspoře času a garanci funkce je tedy v kusové výrobě výhodnější použít některý ze specializovaných obvodů, realizujících obsluhu kapacitní klávesnice. Jedním z nich je MPR121QR2 od výrobce Freescale [20], který byl zvolen do tohoto projektu. Jedná se o druhou generaci obvodů od tohoto výrobce, které obsahují adaptivní vyhodnocovací algoritmy s prvky strojového učení a množství nastavitelných parametrů.



Obrázek 4–9: Tok nasnímaných dat v obvodu MPR121QR2 [20]

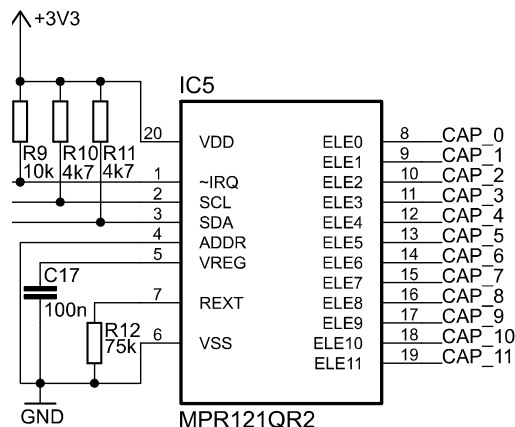
Jak vyplývá z Obrázek 4–9, aplikuje obvod na změřená data (tvořená navzorkovaným napětím na elektrodách v přesně definovaných časech) několik číslicových filtrů s různými časovými konstantami a o stisku tlačítka rozhodne prostřednictvím srovnání s tzv. *baseline* hodnotou. Jedná se o adaptivní mez, která je odvozena z dlouhodobého klidového stavu klávesnice a je s relativně velkými časovými konstantami průběžně přizpůsobována změnám prostředí. Velikost potřebné odchylky naměřených hodnot, charakterizující stisk/uvolnění klávesy, je pro každou elektrodu uživatelsky nastavitelná.

Počet kanálů	12
Napájecí napětí	1,71-3,6 V
Spotřeba	29 $\mu$ A (run mode), 3 $\mu$ A (stop mode)
Komunikační rozhraní	I2C (+ IRQ signál)
Pouzdro	20-vývodové QFN
Cena	1,01 € (20. duben 2013, Mouser, 1 ks)

Tabulka 5 - Základní parametry obvodu MPR121QR2 [20]

Jak vyplývá ze schématu, připojení elektrod je realizováno jednovodičově, bez společného signálu. Hodnoty pasivních součástek pak vycházejí z hodnot katalogových.

Obvod MPR121QR2 komunikuje s nadřazeným systémem pomocí sběrnice I2C. Vývod ADDR, určující adresu obvodu je připojen na nulový potenciál VSS, kdy je pro tuto kombinaci adresa obvodu 0x5A. Pro jiné požadované adresy lze tento vývod spojit s VDD, SDA nebo SCL.



Obrázek 4–10 - Zapojení obvodu obsluhujícího kapacitní klávesnici

Samotné provedení elektrod vychází z aplikační poznámky k tomuto obvodu [23]. Základní pravidla mluví o dodržování co nejkratších a nejtenčích přívodů, o závislosti povolené tloušťky materiálu na ploše elektrody nebo o minimální vzdálenosti tras.

Minimální rozměry elektrody	3 mm x 3 mm
Maximální rozměry elektrody	15 mm x 15 mm
Maximální tloušťka dielektrika	3 mm
Doporučená šířka přívodního spoje	< 0,2 mm
Délka přívodního spoje	1 mm – 200 mm

Tabulka 6 : Doporučené rozměrové parametry dotykových elektrod [23]

V této práci bylo využito všech 12 snímacích kanálů. Obrazec plošného spoje klávesnice je přílohou této práce (elektrody mají rozměr 17 x 17 mm s mezerou 2 mm).

#### 4.2.9 Čtečka RFID karet

Na trhu je široká nabídka čteček RFID nosičů využívajících kmitočet 125kHz. Do tohoto projektu byl zvolen cenově výhodný model EM-18 dodávaný firmou Flajzar [21].



Obrázek 4–11 - Čtečka RFID EM-18 [21]

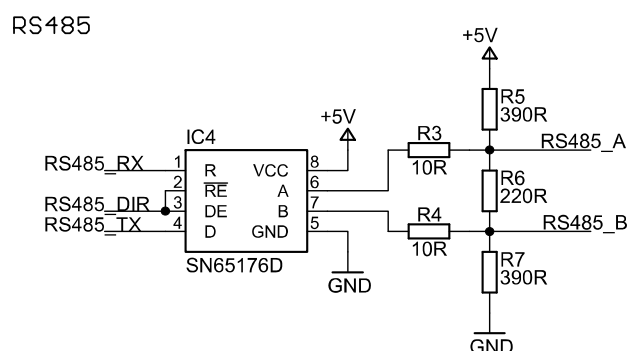
Podporované nosiče	EM 4001 125kHz
Napájecí napětí	5 VDC
Proudový odběr	<50 mA
Čtecí vzdálenost	10 cm
Komunikační rozhraní	Volitelné - UART (9600 8N1) nebo Wiegand26
Cena	353 Kč (20. duben 2013, Flajzar.cz)

Tabulka 7 - Základní parametry čtečky EM-18 [21]

#### 4.2.10 Ostatní rozhraní

Rozšiřující deska kitu MINI2440 v terminálu docházkového systému nabízí krom celků popsaných v předchozích podkapitolách ještě několik dalších rozhraní pro komunikaci s okolím.

Jedním z nich je RS485 pro připojení externího zařízení, například turniketu. Tvoří ho poměrně běžný převodník mezi UART TTL a RS485 SN65176 od výrobce Texas Instruments. Zapojení obsahuje odporový dělič, který kromě zakončení definuje také klidovou úroveň linky.



Obrázek 4–12: Rozhraní RS485

Další součást desky tvoří 8 přípojných bodů, určených pro přídavnou kartu. K dispozici je na nich napájecí napětí 5 V, 3.3 V a 4 GPIO linky (na nichž se nachází volitelně i sériové rozhraní SPI). Možné využití najde například při rozšíření funkcí terminálu o některou z variant bezdrátové komunikace.

Na desce jsou také osazeny výstupy pro dvojbarevnou LED (umístěnou na čelním panelu), napájení a vstup modulu PIR čidla pohybu a též samostatný binární vstup určený pro připojení sabotážního „Tamper“ kontaktu.

## 4.2.11 Shrnutí

Následující obrázek, shrnuje výslednou volbu konkrétních komponent včetně jejich výrobců. Vychází z obecných požadavků na HW vybavu, formulovaných na začátku kapitoly 4.2.



Obrázek 4–13: Souhrn klíčových komponent rozšiřující desky

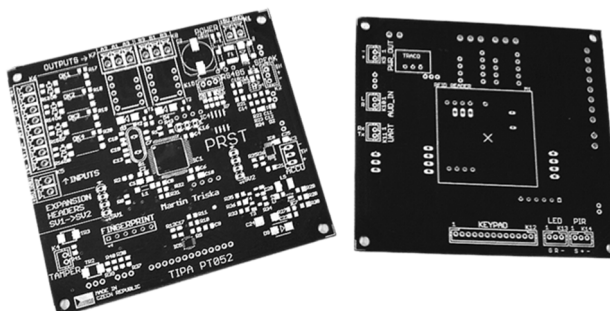
Externí prvky jsou k desce připojeny pomocí konektorů několika typů. Jejich označení a význam uvádí následující tabulka:

Označení	Typ konektoru	Význam
<b>K1</b>	Násuvná svorkovnice	Hlavní přívod napájení
<b>K2</b>	JST-PH	Výstup na reproduktor
<b>K3</b>	PSH	Akumulátor
<b>K4</b>	JST-PH	Sabotážní kontakt
<b>K5</b>	Násuvná svorkovnice	Pomocný výstup napájení
<b>K6</b>	Násuvná svorkovnice	Binární vstupy

<b>K7,K8</b>	Násuvná svorkovnice	Binární reléové výstupy
<b>K9</b>	JST-PH	Napájení kitu MINI2440
<b>K10</b>	JST-PH	Audio vstup
<b>K11</b>	JST-PH	Komunikace s kitem MINI2440
<b>K12</b>	JST-PH	Kapacitní klávesnice
<b>K13</b>	JST-PH	Dvojbarevná LED
<b>K14</b>	JST-PH	Pohybový senzor
<b>K15</b>	JST-PH	RS485
<b>K16</b>	PIN HEADER	SWD rozhraní pro debug MCU
<b>SV1</b>	PIN HEADER	Zásuvný modul – napájení
<b>SV2</b>	PIN HEADER	Zásuvný modul - komunikace

Tabulka 8: Přehled přípojných míst rozšiřující desky

Deska plošných spojů rozšiřující I/O karty je realizována jako dvojvrstvá prokovená. Z důvodu použitých obvodů v miniaturních pouzdrech ji bylo nutné realizovat v VI. třídě přesnosti (šířka spoje/mezery min. 0.15 mm).



Obrázek 4–14: Provedení desky plošných spojů

Deska disponuje černou nepájivou maskou a je opatřena bílým oboustranným servisním potiskem. Za zmínku stojí moderní technologie chemického cínu Ormecon™ CSN. Oproti starším metodám pocínování (například HAL), disponuje Ormecon výhodami, jako je výborná rovinnost povrchu díky minimální tloušťce (0,08μm), bezolovnatost, dlouhá doba skladování, bezproblémové vícenásobné pájení a v neposlední řadě také nižší náklady [22].

### 4.3 Čtečka otisků prstu

Další velice důležitou součást terminálu tvoří vlastní čtečka otisků prstu (v dalším textu bude pro zpřehlednění použita zkratka **FPR**, vycházející z anglických slov *Fingerprint reader*). Vzhledem k využití specifických součástek, jako je optický hranol s CCD snímačem či implementaci komplikovaných algoritmů při zpracování obrazových dat

(jež patří obvykle mezi know-how výrobců) nemá příliš velký význam uvažovat o vlastnoručním vývoji a výrobě FPR.



Obrázek 4–15 - Čtečka otisků Miaxis SM-621 [24]

Pro tento projekt byla zvolena čtečka SM-621 výrobce Miaxis biometrics Co.,Ltd. [24] . Modul FPR se skládá z optické části obsahující zdroj modrého světla, skleněný hranol a samotný snímač. Druhou část tvoří deska plošného spoje se všemi obslužnými obvody: Na této desce se nachází napájecí obvody, sériová paměť flash o kapacitě 4 Mbit pro uchování šablon otisků a DSP obvod řady TMS320 výrobce Texas Instruments. Modul FPR využívá jako komunikační rozhraní sériovou linku UART. Dále je vybaven EN vstupem, který slouží k uvedení zařízení do pohotovostního režimu.

Mezi hlavní parametry čteček otisků (ale i jiných prostředků biometrické identifikace) patří spolehlivostní ukazatelé FRR a FAR. FRR, jakožto zkratka anglických slov *false rejection rate*, vyjadřuje pravděpodobnost odmítnutí oprávněného požadavku na přístup. FAR, vycházející z anglického *false acceptance rate*, značí pravděpodobnost chybného akceptování neoprávněného požadavku. Tyto, ale i ostatní parametry čtečky SM-621 jsou uvedeny v následující tabulce:

<b>Napájecí napětí</b>	<b>3,6 V – 7 V</b>
<b>Pracovní proud</b>	< 100 mA
<b>Špičkový proud</b>	< 120 mA
<b>Sejmutí otisku</b>	< 250 ms
<b>Ověření otisku 1:1</b>	< 600 ms
<b>Vyhledání otisku mezi 240 vzory</b>	< 2 s
<b>Kapacita paměti otisků</b>	240/752/1776 (3 úrovně)
<b>FAR</b>	< 0,001 %
<b>FRR</b>	< 1,5 %
<b>Komunikační rozhraní</b>	UART 57600bps 8N1
<b>Velikost šablony otisku</b>	256 bytů
<b>Rozlišení senzoru</b>	500 DPI

Tabulka 9 - Parametry čtečky Miaxis SM-621[24]

## 4.4 Ostatní prvky

V této podkapitole jsou stručně popsány zbývající komponenty obsažené v terminálu docházkového systému, které jsou připojeny k rozšiřující desce.

**Reproduktor** – Hlavním kritériem pro výběr reproduktoru byla v tomto případě jeho velikost, kdy kvůli prostorovým poměrům v pouzdru nesměl průměr koše překročit 42mm. Požadovaná hlasitost a kvalita reprodukce totiž předem vyřadila z výběru miniaturní reproduktory, používané např. v mobilních telefonech.



Obrázek 4–16: Reproduktor použitý v terminálu

Vybraný reproduktor má zatížitelnost 3 W, membránu z hliníkové slitiny a závěs membrány z gumy. Vzhledem ke své velikosti se vyznačuje poměrně kvalitní reprodukcí, částečně i díky uzavřenému pouzdru terminálu, které tak tvoří ozvučnici.

**Akumulátor** – Zvolen byl akumulátor typu Li-Ion ve standardním rozměru 18650, využívaný například ve svítilnách. Jeho výhodou je nízká cena a snadná dostupnost v různých kapacitách. U velmi levných kusů však skutečná kapacita obvykle nedosahuje kapacity udávané. Zpravidla se pohybuje mezi 2000 mAh – 2700 mAh, což by mělo vystačit přibližně na 2–3 hodiny běžného provozu terminálu.



Obrázek 4–17: Akumulátor Li-Ion 18650

**Senzor pohybu** – Vzhledem k cenám hotových modulů není efektivní vyvíjet pohybový senzor vlastní. Velká část levných PIR modulů (včetně toho, použitého v tomto projektu) je založena na integrovaném obvodu BISS0001, který potřebuje ke své funkci krom samotného čidla minimum externích součástek.



Obrázek 4–18: PIR modul

Pro potřeby terminálu docházkového systému byla dodaná Fresnelova čočka příliš rozměrná a byla nahrazena miniaturním typem QW9705 od dodavatele Flajzar [21].

## 4.5 Pouzdro terminálu

Značnou roli při výběru (a následném nákupu) elektroniky hraje design a zpracování zařízení. I technicky značně vyspělé zařízení nemusí na trhu uspět, pokud bude mít nekvalitní mechanické zpracování a nepraktický či zastaralý design.

V tomto projektu jde o prototyp zařízení, kde by spousta profesionálních postupů, používaných v praxi u větších sérií, nebyla finančně přijatelná. Jedná se například o výrobu pouzdra („krabičky“) na míru, zakázkovou membránovou klávesnici či tamponový potisk čelního panelu. Proto jsou ve značné míře využity technologie realizovatelné i v amatérských podmínkách.

Zvoleno bylo řešení od společnosti Eling.cz [25] dodávající na český trh produkty německého výrobce Bopla, který má v nabídce relativně široké spektrum hliníkových pouzder. Pro potřeby terminálu je vyhovující produktová řada ALU-TOPLINE. Jedná se o dvoudílný hliníkový profil volitelné délky, opatřený plastovými bočnicemi.



Obrázek 4–19: ALU-TOPLINE do výrobce BOPLA [25]

Univerzální pouzdro o šířce 200mm bylo přizpůsobeno aplikaci tohoto projektu částečně strojovým CNC a částečně ručním obráběním. CNC frézování bylo využito z estetických důvodů u čelního panelu, kde je velká část pohledových hran (otvor na čtečku otisků, LCD displej). Ostatní operace jako vrtání děr nebo otvor na kabely ze zadní strany terminálu byly provedeny ručně. Výkres, ze kterého vycházela zdrojová data pro CNC frézku, je přílohou této práce.



Další podstatným krokem k přijatelnému designu terminálu je potisk čelního panelu. Jelikož přímý barevný tisk na hliník, který by byl dostatečně trvanlivý, není při jednom kusu cenově přijatelný, bylo zvoleno řešení v podobě barevného laserového tisku na speciální samolepicí fólii. Jedná se o polyethylenovou folii, odolnou vůči chemikáliím a nečistotám, vhodnou pro průmyslové použití. V bílém matném provedení jí dodává společnost Rayfilm pod výrobním číslem 0503 [26]. Výhoda tohoto řešení spočívá mimo jiné i v překrytí hlav zápusných šroubů na čelním panelu. Otvory do této fólie jsou vyřezány pomocí řezacího plotteru. Zdrojová data pro řezání vychází ze stejné předlohy jako data pro CNC frézku (spolu se samotným grafickým návrhem potisku jsou přílohou této práce), mimo otvorů pro čtečku RFID karet a kapacitní klávesnici – ty zůstávají překryty potiskem.

## 4.6 Změny v HW části

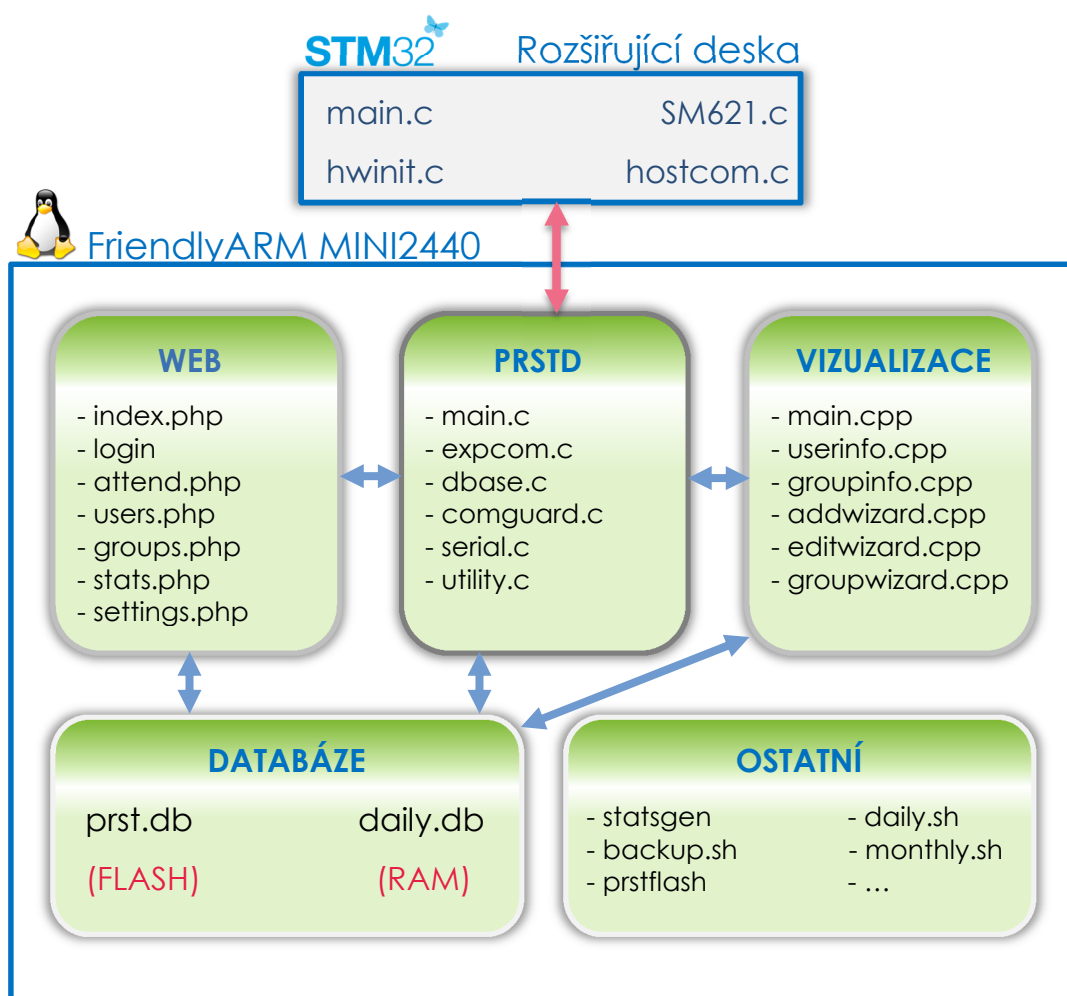
Během osazování DPS a oživování zařízení bylo odhaleno několik dílčích nedostatků či možností lepšího řešení problému. Ty jsou zaneseny ve schématu REV.2, které je elektronickou přílohou této práce. Jejich stručný souhrn včetně navrženého řešení si nyní uvedme:

- Chybná polarita cívek relé RE1 a RE2 – nutno překřížit přívody na DPS
- Nedostatečný proud měniče záložního zdroje – neosazovat rezistor R30 a vývod č. 4 obvodu IC7 připojit přímo na jeho výstup (vývod č. 2) pro deaktivaci proudového omezení
- Nutnost pull-up rezistoru v komunikační lince mezi MINI2440 a rozšiřující deskou (vývod Rx u MINI2440) – doplnit rezistor o hodnotě 4k7 mezi vývod 2 konektoru K11 a napájecí napětí 3,3 V.
- Vhodnost pull-down rezistoru u spínaného měniče pro měření napětí akumulátoru, zamezujícímu jeho spuštění v době nečinnosti terminálu – doplnit rezistor o hodnotě 10k mezi bází T3 a R33.

## 5 SOFTWAREVÁ VÝBAVA

V předchozí části práce, věnující se návrhu hardwaru, byla záměrně vytvořena relativně univerzální platforma, umožňující značnou variabilitu softwarového vybavení. To umožňuje zakázkové přizpůsobení funkcí terminálu dané aplikaci a požadavkům objednavatele. Z tohoto důvodu neexistuje jediné „správné“ nebo nejlepší pojetí softwarové stránky. Programová výbava popisovaná v této části diplomové práce je tedy pouze jakousi ukázkou, demonstrující část možností, jež vytvořená HW platforma nabízí. Z tohoto důvodu jsou nižší vrstvy softwarového modelu pojaty univerzálněji a jimi poskytovaná funkcionality není vždy plně využita nadřazenou vrstvou. Tento přístup značně snižuje čas, potřebný k úpravám a rozšiřováním SW výbavy. Některé části softwaru (např. sloupce databáze) jsou částečně předpřipraveny pro spojení s přístupovým systémem, jehož vývojem se zabývala předchozí práce autora [4].

Na následujícím blokovém znázornění figurují nejdůležitější moduly, z nichž sestává programové vybavení.



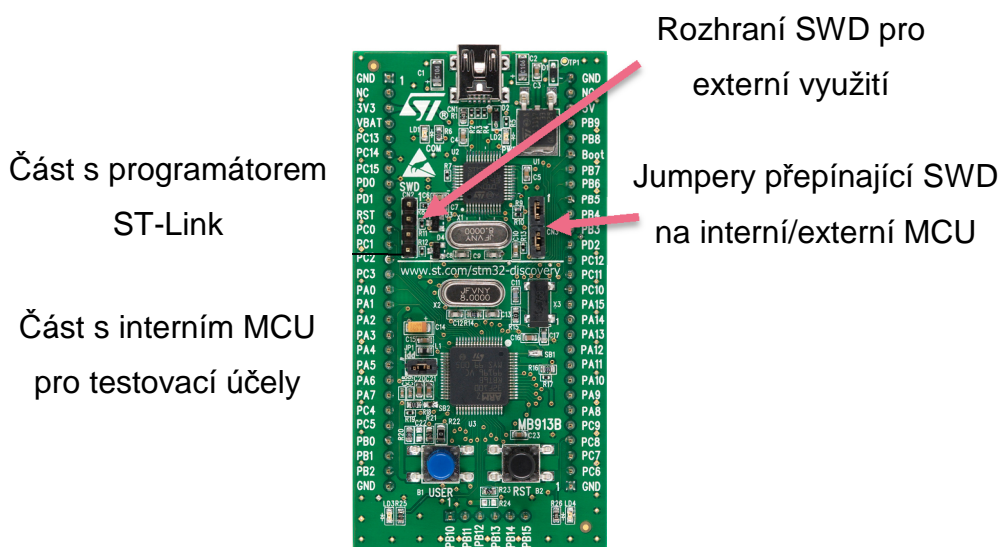
Obrázek 5–1: Nejdůležitější moduly SW výbavy

## 5.1 Rozšiřující deska

Nejnižší místo v hierarchii softwarové výbavy patří obslužnému programu pro rozšiřující desku kitu MINI2440, která je osazena MCU s jádrem ARM Cortex M3 od výrobce STM. Hlavním úkolem tohoto softwaru je zajištění komunikačního mostu mezi aplikacemi pro OS Linux, provozovanými v MINI2440 a periferiemi připojenými k rozšiřující desce.

Vzhledem k tomu, že výrobcem MCU doporučená a podporovaná komerční vývojová prostředí obsahují v bezplatných verzích značná omezení, bylo pro vývoj softwaru zvoleno freewarové prostředí CoCoX CoIDE a překladač GCC [27]. Jedná se o kompletní vývojové prostředí včetně debuggeru a modulu, umožňujícího programování procesoru v aplikaci. Podporována je široká škála procesorů od výrobců jako TI, NXP, Freescale a především prakticky všechny modely od STM z řady STM32Fxxx.

Přenos programu do desky a jeho ladění bylo realizováno prostřednictvím rozhraní SWD (*Serial Wire Debug*). Jedná se o dvojvodičové propojení, které je, jakožto alternativa k rozhraní JTAG, obsaženo v MCU od výrobce STM. Prakticky se spojení s deskou provádí skrze tří pinový konektor K16 s vyvedenými signály SWDIO, SWCLK a GND. Jako protějšek na straně PC je třeba použít hardwarový prostředek označovaný jako ST-LINK. Lze ho zakoupit jako finální výrobek v plastovém pouzdru s rozhraním USB. Jinou možností jeho získání je zakoupení libovolného vývojového kitu z tzv. DISCOVERY řady, které se vyznačují velice nízkou cenou [28] (na prezentačních akcích výrobce jsou tyto kity dokonce rozdávány zdarma). Pro účely této práce byl využit ST-LINK obsažený na kitu STM32VLDISCOVERY. Pomocí jumperů je možno jednoduše zvolit buď propojení ST-LINKu s mikrokontrolérem na kitu, nebo programování externí aplikace.



Obrázek 5–2: STM32VLDISCOVERY jako ST-LINK [28]

Programové vybavení bylo vytvořeno v jazyce C a skládá se z následujících hlavních modulů:

- **main.c** – obsahuje volání inicializačních funkcí a hlavní řídicí smyčku aplikace, která obsahuje úlohy vykonávané s pevným časovým krokem.
- **hwinit.c** – inicializační rutiny pro všechny využívané periferie MCU
- **keyboard.c** – funkce pro obsluhu kapacitní klávesnice
- **SM621.c** – knihovna funkcí pro práci se čtečkou otisků
- **hostcom.c** – funkce pro komunikaci s nadřazeným systémem
- **stm32f10x\_it.c** – obslužné rutiny volané v přerušení
- **hwdefs.h** – definice vývodů MCU a jejich portů + jednoduchá makra pro manipulaci s některými z nich

V následujících podkapitolách budou nejdůležitější funkce těchto modulů popsány.

### 5.1.1 Obsluha čtečky otisků

Knihovna obsahující funkce pro práci se čtečkou otisků prstu (v dalším textu označovanou jako FPR – fingerprint reader) je reprezentována souborem SM621.c. Vytvořena byla pro účely autorovy předchozí práce [4] a do tohoto projektu je s úpravami převzata. Hlavní část knihovny se vyznačuje multiplatformností, nevyskytuje se v ní tedy žádný přímý přístup k hardwaru. Ten zajišťuje volání bytově orientovaných funkcí pro komunikaci po sériové lince:

- `void FPR_putchar(char)`
- `char FPR_getchar(void)`

Komunikace s FPR probíhá po lince typu UART rychlostí 57600 baud bez parity a řízení toku. Základní jednotkou je paket skládající se z proměnného počtu bytů. Pakety mohou být 4 typů – příkazové, datové, koncové a pakety s odpovědí. Příkazový paket slouží k zaslání příkazu do FPR, jeho délka i tvar odpovědi závisí na konkrétním příkazu. Datový paket o pevné délce 128 datových bytů se používá při přenosu „surových“ dat (zpravidla šablon otisků, či sejmutých bitmap) do nebo z FPR. Koncový paket je pak v takovémto přenosu použit jako poslední a nese zbývající data, jeho délka tak není konstantní. Struktura příkazového paketu viz *Tabulka 10*. Paket s odpovědí má strukturu shodnou s paketem příkazovým, avšak namísto kódu požadovaného příkazu obsahuje kód odpovídající úspěchu či chybě. Za ním pak mohou v závislosti na volaném příkazu figurovat další data. Následující funkce pracují na úrovni paketů a interně volají znakově orientované funkce, uvedené výše.

- `char fpr_sendcmd(t_cmd * cmd, int cnt)`
- `char fpr_RecvRAWData(uint8_t* buffer, int cnt)`
- `char fpr_SendRAWData(uint8_t* buffer, int cnt)`

1 byte	4 bytes	2 bytes	1 byte	... ..				2 bytes
Packet Flag01	Equipment Address Code	Packet Length	Command	Parameter 1	Parameter 2	...	Parameter n	Check Sum

Tabulka 10: struktura příkazového paketu FPR [24]

Každý paket začíná i končí znakem 0xC0 (tzv. *Border mask*), jenž slouží k synchronizaci přijímače a vysílače. Kódováním je totiž garantováno, že se tento znak neobjeví uvnitř paketu. Samotný paket se skládá z následujících částí:

- **PacketFlag** – označuje typ paketu (příkazový datový, koncový či odpovědní)
- **AdressCode** – čtečka tento parametr zatím ignoruje, posílají se 4 znaky 0x00
- **PacketLength** – součet bytů v částech Command a Parameters
- **Command** – samotný kód příkazu
- **Parameters** – předávané parametry, závisí na konkrétním příkazu
- **Checksum** – kontrolní součet počítaný přes celý paket

Všechny pakety je nutno kódovat za účelem zamezení výskytu znaku *Border mask* (0xC0) kdekoli uvnitř paketu. Kódovací pravidla jsou následující:

- Pokud paket obsahuje znak 0xC0, je nahrazen dvěma znaky – 0xDB a 0xDC
- Pokud paket obsahuje znak 0xDB, přidá se za něj navíc znak 0xDD

Kompletní seznam podporovaných příkazů včetně popisu chybových kódů je uveden v [24], podrobnější informace ke knihovně pak v [4]. Níže jsou stručně vysvětleny funkce, které jsou v tomto projektu využity. Pokud není uvedeno jinak, vrací v případě úspěšného vykonání nulu a v případě chyby pak její kód.

```
char fpr_CheckFinger(void)
```

Funkce detekuje prst přiložený ke čtečce. V pozitivním případě vrací nulu.

```
char fpr_GetImage(void)
```

Funkce sejme obraz přiloženého prstu a uloží ho do tzv. *Image Bufferu*.

```
char fpr_GenTemplate(char buffer)
```

Z obrazu prstu uloženého v *Image Bufferu* generuje charakteristická data, ta uloží do zvoleného bufferu.

```
char fpr_Search(char buffer, int from, int to)
```

Vyhledá v paměti čtečky otisk, jehož charakteristická data jsou načtena ve zvoleném bufferu (např. pomocí funkce `GenTemplate()`). Zbývající parametry určují prohledávanou část paměti. V případě nalezení vrací pozici otisku v paměti.

```
char fpr_UploadTemplate(char buffer)
```

Vyčte z FPR šablonu otisku, dočasně uloženou ve zvoleném bufferu. Velikost šablony je 256 bytů (po dekódování), přičemž v tomto projektu je za účelem snadnější manipulace ukládána a zpracovávána včetně režijních dat příslušných paketů, kdy je celková velikost 278 bytů.

```
char fpr_DownloadTemplate(char buffer)
```

Slouží pro download šablony otisku do FPR, kde je dočasně uložena ve zvoleném bufferu a připravena pro další operace. Samotná data šablony je po zavolání této funkce třeba odeslat pomocí výše uvedené funkce `fpr_SendRAWData()`.

```
char fpr_Merge(void)
```

Spojí dva načtené otisky prstu a generuje charakteristickou šablonu (kterou je možno následně funkcí `fpr_UploadTemplate()` vyčíst z modulu FPR).

```
char fpr_Delete(int pageNR)
```

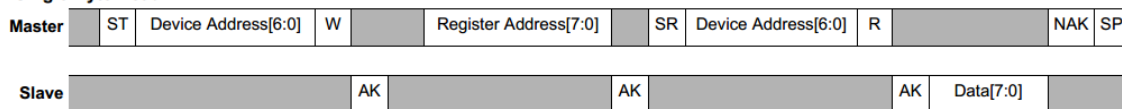
```
char fpr_Store(char buffer, int pageNR)
```

Tyto dvě funkce slouží pro uložení/smazání šablony otisku v paměti FPR, společným parametrem je pozice příslušné paměťové stránky.

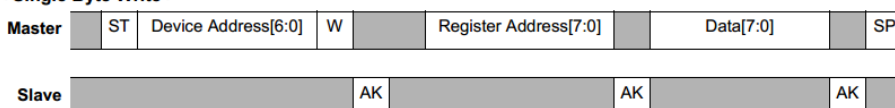
## 5.1.2 Obsluha kapacitní klávesnice

Použitý řadič kapacitní klávesnice komunikuje prostřednictvím rozhraní I2C. Jedná se o typickou master – slave komunikaci, kterou musí zahájit vždy MCU. Obvod disponuje také signálem  $\sim\text{IRQ}$ , umožňujícím vyvolat přerušení v případě stisku klávesy. Tato funkce však není v knihovně využita. Princip komunikace je obdobný jako u jiných IO pro sběrnici I2C a vyplývá z průběhů na *Obrázek 5–3*. Adresu udává způsob zapojení obvodu, viz *Obrázek 4–10*.

### < Single Byte Read >



### < Single Byte Write >



Obrázek 5–3: Časová posloupnost komunikace s *MPR121QR2* [20]

MPR121 obsahuje 128 osmibitových registrů, z nichž jedna část uchovává aktuální provozní hodnoty a naměřená data a druhá část slouží k nastavení parametrů snímání. 36 registrů z druhé skupiny je inicializováno na výrobcem doporučené hodnoty prostřednictvím funkce `KEYB_Init()`, která je popsána níže.

Funkce spojené s obsluhou obvodu MPR121QR2 se nachází v soboru `keyboard.c`. Samotnou nízkoúrovňovou komunikaci s obvodem podle výše uvedených časových diagramů zajišťují 2 lokální funkce:

- `unsigned char Keyb_Read_REG(unsigned char addr)` pro čtení hodnoty registru na adrese `addr`
- `void Touch_Write_REG(unsigned char addr, unsigned char data)` pro zápis hodnoty `data` do registru na adrese `addr`.

Pro použití v aplikaci jsou k dispozici funkce, které si nyní popíšeme:

```
void KEYB_Init(void)
```

Funkce provede inicializaci obvodu MPR121QR2 na výchozí hodnoty a uvede ho do aktivního režimu. Vzhledem k širokým možnostem nastavení, které obvod nabízí, byla ke stanovení výchozích hodnot využita aplikační poznámka, poskytovaná výrobcem [29]. S těmito hodnotami funguje instalovaná kapacitní klávesnice bez problémů, není tedy důvod je měnit. Inicializace ovlivňuje tyto skupiny registrů:

- Parametry tzv. *baseline* - adaptivní reference, která zajišťuje dlouhodobou přesnost snímání i při postupných změnách okolního prostředí
- Rozhodovací meze pro vyhodnocení stisku/uvolnění jednotlivých kláves
- Vzorkovací perioda (nastavena na 16 ms) a počet vzorků pro průměrování (4)
- Zapnutí automatické konfigurace a re-konfigurace
- Spuštění vzorkování na zvolených vstupech (v tomto případě na všech)

Tuto funkci je třeba zavolat po spuštění zařízení, avšak až ve chvíli, kdy jsou již inicializovány potřebné periferie MCU (hodiny, GPIO, I2C).

Vzhledem k faktu, že po prvním spuštění probíhá adaptace hodnoty *baseline* na aktuální pracovní prostředí klávesnice (viz kapitola 4.2.8), nelze klávesnici po určité době (cca 20 sekund) používat. Toto chování však není na závadu, protože během adaptace probíhá zpravidla také bootování OS nadřazeného systému.

```
uint8_t KEYB_ReadKey(void)
```

Funkce slouží pro čtení aktuálně stisknuté klávesy. Je vhodné ji volat periodicky z hlavní smyčky (tzv. *polling*). Pokud není stisknuta žádná klávesa, vrací funkce nulu. V opačném případě závisí návratová hodnota na nastaveném režimu klávesnice. V číselném módu je vrácena číslice ve formátu ASCII, v alfanumerickém pak písmeno odpovídající počtu

stisků dané klávesy (obdobně jako při psaní SMS na klasickém mobilním telefonu). Podporovány jsou i české znaky s diakritikou, které jsou kódovány znakovou sadou windows-1250.

```
void KEYB_SetMode(uint8_t setmode)
```

Funkce nastavuje požadovaný režim klávesnice. Její parametr může nabývat hodnot `keybmode_numeric` pro číselný, nebo `keybmode_alpha` pro alfanumerický režim. Současně je při změně na režim alfanumerický aktivován tzv. auto-repeat, který umožňuje výběr požadovaného písmene držením příslušné klávesy. Funkce nekomunikuje s MPR121QR2 ani nevyužívá periferie, její provedení je tedy vždy úspěšné.

```
uint8_t KEYB_GetMode(void)
```

Funkce vrací aktuálně nastavený režim klávesnice tedy `keybmode_alpha` pro alfanumerický režim nebo `keybmode_numeric` pro číselný režim.

### 5.1.3 Komunikace s nadřazeným systémem

Rozšiřující deska je s nadřazeným systémem spojena jedním obousměrným komunikačním kanálem, tvořeným sériovou linkou typu UART. Komunikace probíhá rychlostí 9600 bps. Byla zvolena relativně nízká přenosová rychlost, jelikož se v současné variantě SW nepředpokládají přenosy větších objemů dat. Rychlost lze v souboru *hwdefs.h* jednoduše změnit úpravou konstanty `HOST_BAUDRATE`. Struktura paketu je popsána v následující tabulce:

BORDER	LENGTH	LENGTH	DATA	CHECKSUM	CHECKSUM
MASK	MSB	LSB		MSB	LSB
1 byte	1 byte	1 byte	n bytes	1 byte	1 byte

Tabulka 11: Struktura paketu komunikace s nadřazeným systémem

Modul *hostcom.c* obsahuje funkce spojené s příjmem i vysíláním dat, které budou v této kapitole popsány. Samotný příjem paketů, realizovaný stavovým automatem se nachází v obslužné rutině přerušení. Stav komunikace indikuje příznak `Host_Com_Stat`. Může nabývat následujících hodnot:

- `PACKET_RECEIVED` – byl přijat kompletní paket
- `PACKET_ERR` – struktura přijatého paketu je poškozena
- `CHECKSUM_ERR` – neodpovídá kontrolní součet přijatého paketu
- `OVERFLOW_ERR` – paket byl přijat před zpracováním předchozího



V asynchronním časovém kroku hlavní smyčky je příznak `Host_Com_Stat` testován. V případě, že došlo k chybě, informuje se o této skutečnosti nadřazený systém. Pokud však byl přijat kompletní nepoškozený paket, je volána funkce `Host_Process()`, jejímž úkolem je paket zpracovat a vykonat příslušnou reakci. Po návratu této funkce je příznak `Host_Com_Stat` resetován a program vyčkává další paket. Zpracovatelskou funkci si nyní popíšeme podrobněji.

```
int Host_Process(void)
```

Funkce na základě přijatého požadavku vykoná příslušný příkaz. Nadřazenému systému pak zašle zprávu o úspěšnosti dokončení ve tvaru:

HEADER	SUCCESS OR ERR. CODE	COMMAND CODE	DATA (optional)	FOOTER
3 bytes	1 byte	1 byte	n bytes	3 bytes

Tabulka 12: Struktura paketu s odpovědí (vychází z obecného tvaru paketu v Tabulka 11)

V případě úspěchu je v prvním bajtu za hlavičkou vrácen kód `HOSTOUT_SEND_ACK`, v případě chyby pak chybový kód. Jejich výpis uvádí hlavičkové soubory *errors.h* a *sm621.h*. V následujícím bajtu je zopakován kód příkazu, kterého se odpověď týká. Další – nepovinnou – část paketu tvoří sekce data, využitá např. při zjišťování stavu binárních vstupů nebo echu.

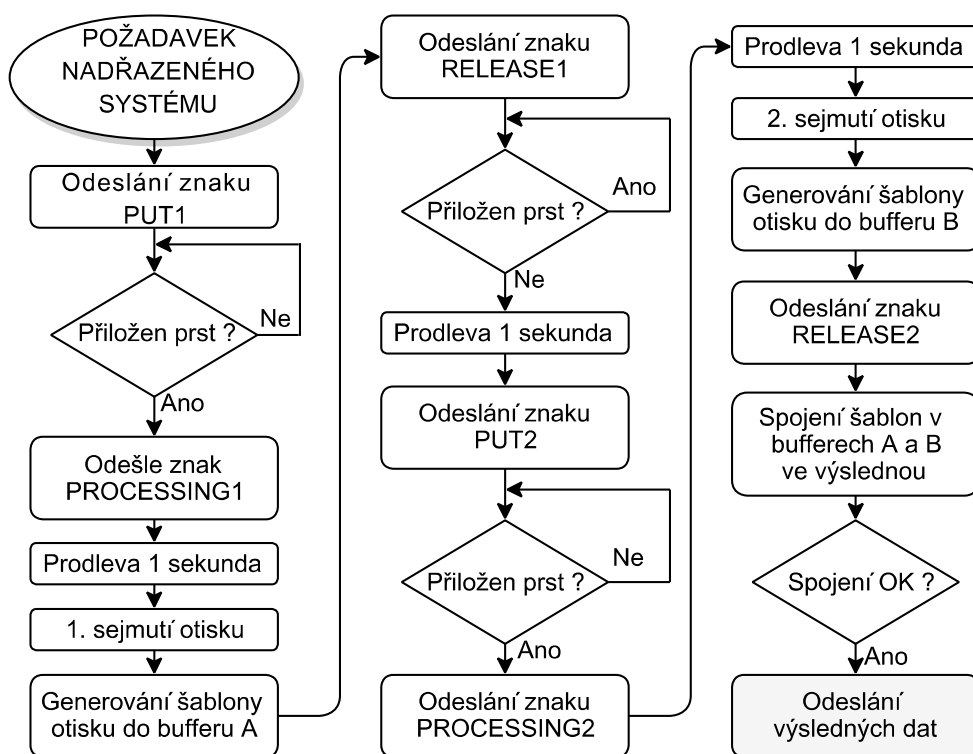
Dle jednobytového kódu příkazu, který je extrahován z příchozího paketu, vykonává funkce následující činnosti:

HOSTIN_ECHO_REQUEST	Vrátí v prvním bajtu datové části informace o aktuálním stavu systému (zdroj napájení, režim...), v další části pak verzi firmware. Funkce nikdy nevrací chybový kód.
HOSTIN_FPR_SWITCH	Slouží v závislosti na parametru k zapnutí či vypnutí čtečky otisků prstu
HOSTIN_KEYBMODE	V závislosti na parametru nastaví alfanumerický nebo číselný režim kapacitní klávesnice
HOSTIN_BOOTLOAD_RESET	Provede příslušné kroky pro vyvolání bootloaderu, který se nachází v systémové části paměti. Poté je možno pomocí nadřazeného systému a skriptu <i>prstflash.sh</i> aktualizovat program v MCU.
HOSTIN_FPR_SCAN	Vyvolá proces scanování otisku, jehož výsledkem je v případě úspěšného dokončení šablona otisku prstu, která je odeslána do nadřazeného systému.

HOSTIN_FPR_DOWNLOAD	Uloží na stanovenou pozici v modulu FPR šablonu otisku prstu. Požadovaná pozice a data šablony se nachází v datové části příkazu.
HOSTIN_FPR_DELETE	Smaže šablonu otisku prstu v modulu FPR na pozici udané v datové části paketu.
HOSTIN_FPR_AMPEN	Slouží k permanentnímu zapnutí audio zesilovače, které tak není ovlivněno přechodem do úsporného režimu.
HOSTIN_RDBIN	Vrátí aktuální hodnoty binárních vstupů.
HOSTIN_WRBIN	Nastaví do požadovaného stavu reléové výstupy.

Tabulka 13: Význam kódů příchozích příkazů

Požadavek HOSTIN\_FPR\_SCAN vyvolá přechod do speciálního režimu, který slouží k získání šablony otisku. Je nutné nasnímat dva otisky téhož prstu a provést jejich sloučení. O stavu, v němž se proces nachází, je informován nadřazený systém následujícím způsobem:



Obrázek 5-4: Proces tvorby šablony otisku prstu

Nyní se z příchozí komunikace přesuňme na stručný popis komunikace odchozí. Ta je realizována samostatnými funkcemi, volanými při zachycení příslušné události. Všechny dále uvedené funkce využívají interně obecnou funkci pro odesílání paketu do nadřazeného systému `Host_send(uint8_t* data, uint16_t length)`. Z důvodu úspory prostoru budou namísto kompletních hlaviček funkcí uvedeny pouze jejich názvy (bez prefixu `Host_`)

Send_ScannedFPR( )	Funkce volaná po sejmutí otisku prstu. Pokud je otisk v databázi FPR, odešle jeho pozici, v opačném případě HOSTOUT_FPR_NOTFOUND
Send_PressedKey( )	V případě stisku klávesy na kapacitní klávesnici odešle v závislosti na nastaveném režimu (alfanumerický, číselný) hodnotu příslušné klávesy.
Send_ReadRFID( )	Při přiložení RFID nosiče odešle jeho identifikátor
Send_ScanProgress( )	Funkce během procesu snímání otisku za účelem získání šablony odesílá informace o průběhu akce.
Send_PWRChange( )	Při výpadku napájení ze sítě / opětovném připojení k síti odesílá informaci o aktuálně využívaném zdroji energie. Upozorňuje též na téměř vybitý akumulátor.
Send_ActiveMode( )	Odesílá informaci o přechodu z úsporného režimu, způsobeném zachyceným pohybem resp. přechodu opačným, způsobeném delší neaktivitou.
Send_Echo( )	Odešle stanovená data jako reakci na požadavek na echo z nadřazeného systému.
Send_Ack( )	Odešle výsledek vykonání příkazu iniciovaného z nadřazeného systému.

Tabulka 14: Seznam funkcí pro odchozí komunikaci

## 5.2 Aplikační server

Vzhledem k tomu, že vývojový kit MINI2440 disponuje operačním systémem umožňujícím paralelní běh více procesů, bylo vhodné vytvořit jeden hlavní program – proces ve formě serveru. Ten zajišťuje nejdůležitější funkce systému a je propojen s dalšími příslušnými procesy. U takto vytvořeného hlavního procesu se předpokládá jeho nepřetržitý běh po celou dobu provozu terminálu (s výjimkou aktualizace SW výbavy, kterou je tak vhodné plánovat např. do pozdních nočních hodin).

Pro účely ukázkové SW výbavy docházkového terminálu byl vytvořen program s názvem prstd. Je spouštěn vždy při startu terminálu a mezi jeho nejdůležitější činnosti patří:

- Logování přístupů – nejdůležitější funkce docházkového systému
- Komunikace s podřízenou rozšiřující deskou skrze UART
  - Reakce na HW událost (např. probuzení z úsporného režimu)
  - Zprostředkování rozhraní pro ostatní procesy v MINI2440 (např. smazání otisku z FPR prostřednictvím webové aplikace)
- Spuštění procesu zajišťujícího vizualizaci

OS Linux nabízí pro předávání dat mezi procesy řadu prostředků, mezi něž patří např. systém zpráv, sdílená paměť, nebo anonymní či pojmenované roury. Pro účely tohoto projektu byla zvolena poslední možnost – pojmenované roury. Hlavním důvodem bylo multiplatformní prostředí Qt, v němž je vytvořeno grafické rozhraní [30]. To zapouzdřuje standardní prostředky (API) operačního systému do vlastních funkcí a datových typů. Přímé využití Linuxové IPC (*Inter-process communication*) v QT aplikaci by tak bylo v rozporu s touto filozofií. Vzhledem k tomu, že se s pojmenovanou rourou pracuje obdobně jako s běžnými soubory – pomocí shodných funkcí pro čtení a zápis, nedochází zde ke kolizi.

Další výhodou tohoto řešení je i fakt, že operační systém garantuje atomicitu přenosu, tzn., pokud do roury zapíšou dva procesy naráz, nedojde k porušení integrity dat. Tato vlastnost se však vztahuje pouze na zprávy do maximální velikosti dané konstantou `PIPE_BUF`, definovanou v souboru *limits.h*. Tato hodnota je v našem případě 4096 bytů, což je pro současné využití naprosto dostačující.

### 5.2.1 Funkcionalita

Hlavní část programu `prstd` je tvořena dvěma paralelně běžícími vlákny. První z nich pomocí funkce `epoll()` zachytává požadavky ze dvou zdrojů – rozšiřující desky prostřednictvím sériové linky (reprezentované souborem `/dev/ttySAC1`) a z pojmenované roury (soubor `/var/pipes/prstd_in`), která slouží jako rozhraní ostatním procesům. Druhé vlákno pak tyto požadavky vykonává. Jako mezipaměť žádostí slouží kruhový buffer o kapacitě 5 požadavků. Přístup k němu je synchronizován pomocí dvou IPC objektů typu semafor (tzv. problematika producent-konzument). Program dále využívá logování do standardního systémového logu (log je k dispozici v souboru `/var/log/messages`).

Jak již bylo uvedeno, program komunikuje s okolím skrze dva hlavní vstupní body (sériovou linku, rouru `/var/pipes/prstd_in`) a jeden bod výstupní (rouru `/var/pipes/prstd_out`). Protokol využívaný sériovou linkou je binární a byl popsán v kapitole 5.1.3. Komunikace skrze roury využívá velice jednoduchý ASCII protokol. U vstupní roury je příkaz reprezentován písmenem abecedy, následovaným případným parametrem. Data ve výstupní rouře jsou za účelem zjednodušení vyhodnocení v programu vizualizace opatřena speciálními ASCII symboly, reprezentujícími jejich význam. Nyní si stručně popíšeme podporované požadavky a příslušné reakce na ně:

Příchozí požadavky ze sériové linky (od rozšiřující desky)	
Požadavek	Reakce
ACTIVE (detekován pohyb)	Zapne či vypne podsvícení LCD displeje zápisem „1“ nebo „0“ do <code>/dev/backlight</code> .
SCANNED_FPR (načten otisk)	Pokud otisk patří existujícímu uživateli, zapíše přístup do databáze. Vizualizaci předá práva, jméno a příjmení daného uživatele.
CARD_READ (načtena karta)	Předá číslo karty vizualizaci (autentifikace kartou není v současné verzi aktivní).
SCAN_PROGGRES	Předá aktuální stav procesu snímání otisku vizualizaci.
KEY_PRESSED (stisknuta klávesa)	Předá hodnotu stisknuté klávesy vizualizaci, pokud se jedná o modifikátor přístupu (např. oběd, pauza...), nastaví ho jako aktivní pro následující přístup. Pokud však do 5 sekund nedojde k identifikaci uživatele pomocí otisku, je nastavený příznak smazán.
POWER_CHANGE (změna v napájení)	Při přechodu na záložní zdroj je pomocí skriptu <code>pwrbackup.sh</code> převedena denní databáze z RAM do FLASH. Při nízkém stavu akumulátoru je systém vypnut příkazem <code>shutdown</code> .
ACK / ERR	Modul <code>commguard.c</code> vyhodnocuje dle návratové hodnoty úspěšnost vykonání příkazu, zaslaného rozšiřující desce a loguje případné chyby.

Příchozí požadavky z roury (z vizualizace nebo webového rozhraní)	
FPR_SWITCH	Zapne nebo vypne FPR
TEMPLATE_SCAN	Aktivuje proces snímání otisku za účelem vytvoření šablony, která je následně uložena do dočasného souboru <code>/var/tmp/teplate</code>
TEMPLATE_STORE	Šablonu otisku prstu, sejmoutou příkazem <code>TEMPLATE_SCAN</code> uloží na požadovanou pozici ve FPR.
TEMPLATE_DELETE	Ve FPR smaže šablonu otisku na příslušné pozici.
KEYBMODE_SET	Nastaví režim klávesnice. Možnosti – číselný, alfanumerický, nebo reprezentující modifikátory
BOOTLOAR_RQ	Vyvolá restart rozšiřující desky s následným skokem na bootloader.

Tabulka 15: Souhrn příchozích požadavků pro aplikaci *prstd*

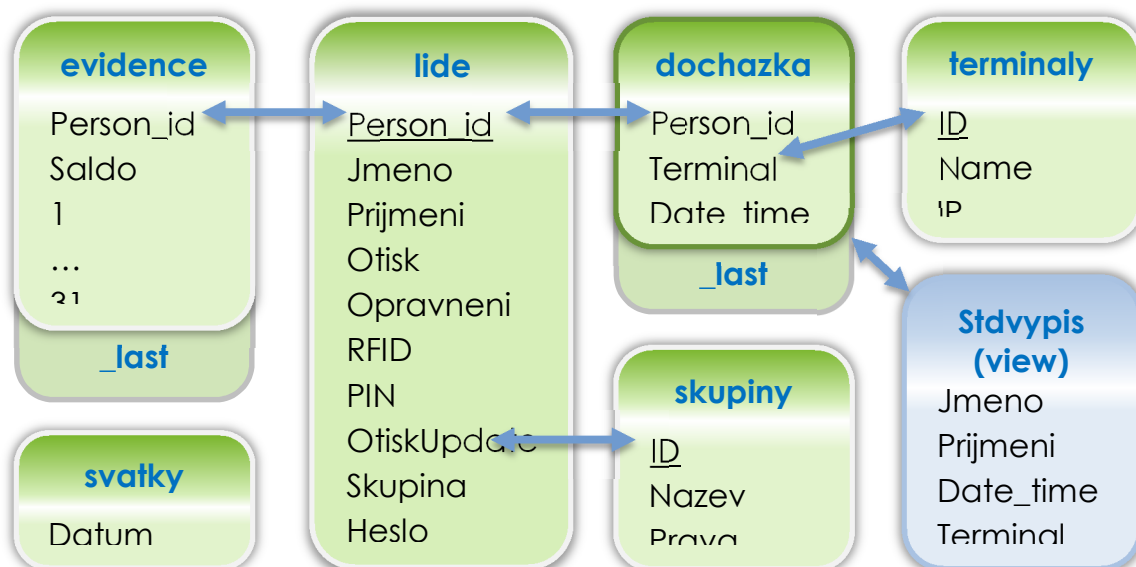
## 5.3 Databáze

Jako hlavní úložiště dat v systému byla zvolena databáze typu SQLite. Jedná se o databázový systém, který je vhodný pro embedded systémy pro svou nenáročnost na hardwarové prostředky. Tento databázový systém pokrývá naprostou většinu z funkcí dotazovacího jazyka SQL. Velkou výhodou pro tento projekt je i přímá podpora SQLite v PHP a Qt.

Knihovna, vhodná pro přímé využití v programech psaných v jazyku C však není standardní součástí OS Linux na kitu MINI2440. Bylo nutné stáhnout zdrojové kódy z domovské stránky SQLite [32] a následně je přeložit překladačem GCC, určeným pro procesory ARM. Krom knihovny je výsledkem překladače také samostatně spustitelný program *sqlite3*. Ten umožňuje pracovat s databází z prostředí příkazového interpretu – shellu. Využít ho lze například z automatizovaně spouštěných skriptů nebo CGI skriptů.

### 5.3.1 Struktura

Aktivní databáze v terminálu docházkového systému je reprezentována dvěma soubory. Hlavní část tvoří soubor *prst.db*, umístěný na paměťové kartě typu SD osazené v kitu MINI2440. V této databázi jsou obsaženy krom všech ostatních dat i kompletní údaje o docházce za předchozí a aktuální měsíc, mimo záznamy z aktuálního dne. Ty se nachází v databázi *daily.db* umístěné v paměti RAM (v adresáři */var/*). Takto rozdělené uložení přináší dvě výhody. Nejčastěji prováděná operace – zápis záznamu o přístupu probíhá rychleji a s menší systémovou režii do operační paměti, než do externí paměti SD karty. Další benefit vyplývá ze značně nižší frekvence zápisů na SD kartu, což prodlužuje její životnost a tím zvyšuje celkovou spolehlivost terminálu.



Obrázek 5–5: Struktura hlavní databáze *prst.db*

Strukturu hlavní databáze na SD kartě znázorňuje Obrázek 5–5. Nyní si stručně popíšme význam jednotlivých tabulek a některých řádků v nich obsažených:

- **dochazka** – slouží jako hlavní úložiště záznamů o docházce, které jsou importovány vždy s koncem dne z databáze *daily.db*. S každým ukončeným měsícem je obsah tabulky docházka přesunut do tabulky *dochazka\_last*.
- **evidence** – tabulka obsahuje odpracovanou dobu zaměstnanců za jednotlivé dny v aktuálním měsíci a její celkovou bilanci, označovanou jako saldo. Obdobně jako u tabulky *dochazka* se pro účely zpětného zpracování obsah jeden měsíc archivuje (přesunem do *evidence\_last*)
- **lide** – obsahuje centrální databázi uživatelů. Ve sloupci *Otisk* je uložena šablona otisku prstu v hexadecimální reprezentaci. Sloupec *Opraveni* obsahuje znak určující práva dané osoby, přičemž může nabývat hodnot U – uživatel, G – skupinový správce nebo A – hlavní správce. Datum poslední změny otisku je uloženo ve sloupci *OtiskUpdate* a hash hesla (typu sha-1) do webového rozhraní ve sloupci *Heslo*.
- **terminaly** – v této práci je tabulka využita pouze jako seznam modifikátorů odchodů (např. lékař, oběd). Sloupec IP je určen pro adresy terminálů v případě provázanosti s přístupovým systémem.
- **skupiny** – záznamy v tabulce reprezentují jednotlivá oddělení ve firmě. Pro případ provázání s přístupovým systémem je v tabulce sloupec přístupových oprávnění *Prava*. Funkcionalita však není v tomto projektu využita.
- **svatky** – obsahuje seznam dnů pracovního klidu, který využívá program statsgen při výpočtu bilance pracovní doby – salda.
- **stdvypis** – jedná se o pohled na tabulku *dochazka*, v němž je za sloupec *Person\_id* dosazeno příjmení a jméno uživatele a za ID typu odchodu jeho plný název. Pohled využívá například webové rozhraní.

V databázi jsou dále naprogramovány triggery, které pomáhají automatizovat databázové operace spojené s editací uživatelů či skupin. Zabezpečují tyto činnosti:

- Při smazání uživatele z tabulky *lide* jsou smazány příslušné údaje s docházkou a pracovní dobou
- Při přidání uživatele je provedeno vložení řádku do tabulky *evidence*
- Při mazání skupiny je zajištěno, aby v číslování nebyly mezery – tzn. U skupin s vyšším ID je to jejich o jedničku zmenšeno. Adekvátně jsou pak upraveny i čísla skupiny u jednotlivých uživatelů v tabulce *lide*.

Databáze v RAM a FLASH jsou synchronizovány jedenkrát za den společně s generováním statistických dat (viz kapitola 5.6.1). V případě výpadku síťového napájení se však databáze synchronizují okamžitě a případné nové záznamy jsou přesměrovány přímo do hlavního souboru *prst.db*. V případě potřeby lze obě databáze v rámci jednoho

otevřeného připojení sloučit SQL příkazem ATTACH DATABASE, což výrazně zjednoduší provádění některých operací (např. výpis docházky na webu).

### 5.3.2 Výkon

Výkon databáze byl testován na testovací databázi s 50 000 záznamy v tabulce *dochazka* a stejným počtem i v tabulce *dochazka\_last*. Počet uživatelů (řádků v tabulce *lide*) byl 100. Velikost souboru s touto databází pak činila 3 357 kB. Jak patrně z tabulky, je nejnáročnější operací jakékoliv vyhledávání ve velkém počtu údajů. Naopak pouhý výpis naposledy přidaných záznamů do tabulky zabere relativně málo času. Velikost testovací databáze je záměrně vyšší, než jaká se předpokládá při typickém nasazení, aby bylo možno podchytit případné problémy s nedostatkem systémových prostředků (např. paměti RAM) během zpracování dotazů.

Operace	Střední hodnota doby vykonání
Výběr přístupu v rozmezí kalendářních dat, tabulka obsahuje jeden odpovídající záznam na pozici 30 000.	<b>704 ms</b>
Výběr přístupu v rozmezí kalendářních dat, tabulka neobsahuje žádný odpovídající záznam.	<b>1064 ms</b>
Výběr přístupů podle ID uživatele, 20 odpovídajících záznamů v tabulce	<b>390 ms</b>
Výpis posledních 30 záznamů o docházce prostřednictvím pohledu stdvypis	<b>18 ms</b>
Výpis posledních 30 záznamů o docházce z tabulky dochazka	<b>12 ms</b>
Výpis denní docházky (100 záznamů) z databáze <i>daily.db</i>	<b>29 ms</b>

Tabulka 16: Doba vykonávání některých databázových dotazů

Uvedené hodnoty samozřejmě vykazovaly jistý rozptyl, avšak nebyl výrazný, a byl tak zanedbán. Měření probíhalo na systému se spuštěnými základními aplikacemi (prstd, vizualizace). I přesto, že odhadovaný objem databáze v systému (viz kapitola 3.1) bude pravděpodobně znatelně nižší, je výhodné provádět operace vyžadující vyšší množství databázových požadavků, jako je generování statistik (viz kapitola 5.6.1), v době nižšího vytížení systému, například v nočních hodinách.



## 5.4 Vizualizace pro LCD displej

Grafické uživatelské rozhraní (dále označované jako GUI – Graphical User Interface) realizované pomocí 3,5“ displeje slouží především k základnímu informování uživatelů při každodenním používání terminálu. Ačkoliv vytvořený software umožňuje provádět z GUI i administrativní funkce, například přidávání a editaci uživatelů nebo manipulaci se skupinami, je výrazně komfortnější a rychlejší vykonávat tyto činnosti přes webové rozhraní. Jedinou editační činností, která využití GUI vyžaduje, je sejmutí otisku prstu.

### 5.4.1 Přímý přístup k LCD

Použitý grafický barevný displej má rozlišení 320 x 240 bodů (označované jako QWGA). V OS Linux je reprezentován souborem `/dev/fb0`. Do tohoto souboru lze přímo zapisovat grafická data v takzvaném formátu RGB565, kdy každému pixelu odpovídá 16 bitů – 2 bajty. Displej je tedy schopen zobrazit 65 536 barev. Jednotlivé bity dvojбайtového slova jsou barvám přiřazeny takto:

- [0 : 4] – modrá barva
- [5 : 10] – zelená barva
- [11 : 15] – červená barva

Přímý přístup k framebufferu využívá v tomto projektu bootovací sekvence, kdy se po dobu inicializace a spouštění aplikací zobrazuje logo a nápis „Čekejte prosím“. Tato bootovací obrazovka je, jakožto bitmapa ve formátu RGB565 uložena v souboru, který lze předat displeji k zobrazení:

```
cat /sdcard/bootlogo.bin > /dev/fb0
```

Ze souboru `/dev/fb0` lze samozřejmě také číst, což může být užitečné v případě, kdy je třeba vygenerovat snímek aktuálního zobrazení na LCD, tzv. *screenshot*. Tento přístup byl využit i pro snímky displeje, prezentované touto prací.

### 5.4.2 Prostředí Qt

Pro tvorbu aplikace zajišťující grafické uživatelské rozhraní na LCD terminálu bylo zvoleno využití knihoven Qt. Jedná se o multiplatformní prostředí, určené pro kompletní vývoj GUI aplikací. Šířeno je pod tzv. duální licenci – jakožto open source pod GPL v3 a LGPL v2, nebo jako komerční pod Qt commercial license. Vedle operačních systémů, provozovaných převážně na desktopových platformách (Mac OS X, Windows), jsou podporovány také další OS, jako VxWorks, Windows Embedded a především Linux on Arm [32]. Aktuální verze byla v době psaní této práce 5.0.2.

Hlavním programovacím jazykem, používaným při vývoji Qt aplikací je C++, v němž je vytvořen i program zajišťující GUI v této aplikaci. Kit MINI2440 obsahuje rozhraní Qt

již od výrobce, avšak ve značně zastaralé verzi 2.2.0. Pro tento projekt byla využita verze 4.6.3, která byla zkompileována pomocí překladače GCC pro platformu ARM ve verzi 4.4.3. V konfiguračních souborech je třeba před samotnou kompilací provést několik změn, které jsou popsány např. v [33]. Další krok tvoří konfigurace, při níž se pomocí parametrů předaných skriptu *configure* dají nastavit požadované vlastnosti a zvolit moduly, které budou ve výsledných knihovnách k dispozici. Vzhledem k množství voleb a možnosti jejich úprav byl vytvořen samostatný skript *konfigurace.sh* jako wrapper skriptu původního.

Qt je velmi komplexní prostředí pro vývoj aplikací, z něhož je pro účely tohoto projektu využita jen malá část (objektů, reprezentovaných třídami v C++, jsou v Qt k dispozici řádově stovky). Použity byly objekty z těchto knihoven:

- **libQtCore.so.4.6.3** – obsahuje třídy, které přímo nesouvisí s grafickými prvky, jedná se například o synchronizační objekty, časovače nebo souborové operace
- **libQtGui.so.4.6.3** – zahrnuje všechny grafické prvky, jako jsou tlačítka, checkboxy (souhrnně označované jako tzv. *widgety*) a též např. pomocné objekty, usnadňující zarovnání prvků do formulářů
- **libQtSql.so.4.6.3** – slouží pro implementaci spojení s databází, podporuje více typů databází, např. MySQL nebo SQLite

### 5.4.3 Funkce

GUI je v terminálu docházkového systému realizováno programem *vizualizace*. Ten se spouští po startu systému z hlavního procesu *prstd* (viz kapitola 5.2). Po dobu běhu komunikuje s tímto nadřazeným programem prostřednictvím pojmenovaných rour. V GUI jsou využity volně šiřitelné obrázkové ikony z [39]. Implementovanou funkcionalitu nejlépe ilustruje výpis modulů (reprezentovaných stejnojmennými třídami):

- **Vizualizace.cpp** – hlavní třída, realizující základní obrazovku, zpracování dat z roury a práci s ostatními třídami
- **Addwizard.cpp** – průvodce přidáním uživatele. Obrazovka umožňující sejmutí otisku prstu je využita i v průvodci editací.
- **Editwizard.cpp** – průvodce editací uživatele.
- **Groupwizard.cpp** – průvodce editací skupin
- **Menu.cpp** – obrazovka, umožňující v závislosti na oprávnění uživatele vyvolat některý z výše uvedených průvodců
- **Analogclock.cpp** – třída, zajišťující hodiny analogového typu na základní obrazovce. S úpravami je převzata z ukázkových kódů prostředí Qt.

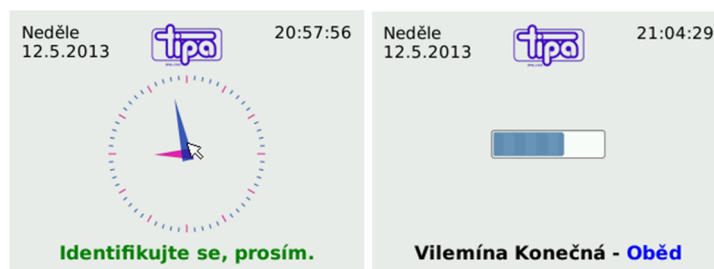
Dále program obsahuje pomocné moduly, které se přímo nepodílí na GUI, ale jsou nezbytné pro správnou funkci:

- **Userinfo.cpp** – zapouzdřuje informace o konkrétním uživateli a obsahuje metody pro práci s nimi. Zajišťuje komunikaci s databází. Instanci této třídy využívají průvodci pro editaci a přidání uživatele.
- **Groupinfo.cpp** – obdobný význam jako userinfo.cpp, avšak ve vztahu ke skupinám.
- **Utility.cpp** – obecné pomocné funkce, zajišťující např. generování záhlaví průvodců, nebo převod českých znaků z kapacitní klávesnice na kódování unicode, využívané GUI i databází.

V následujícím textu si stručně popíšeme chování a možnosti jednotlivých částí GUI. Tuto část tak lze chápat také jako rychlý návod k použití základních funkcí vytvořené softwarové výbavy. Publikované snímky displeje byly vytvořeny utilitou *snapshot* (obsaženou v kitu MINI2440 již z výroby), která přečte framebuffer a uloží jeho obsah jako obrázek ve formátu .png.

### Základní obrazovka

Je zobrazena na displeji prakticky po celou dobu chodu zařízení. Největší plochu zabírají „analogové“ kruhové hodiny v centrální části obrazovky (zdrojový kód hodin byl s úpravami převzat z ukázkových kódů, dodávaných s prostředím Qt). Ve vrchní části se nachází datum, logo firmy a čas, ve spodní pak stavový řádek.



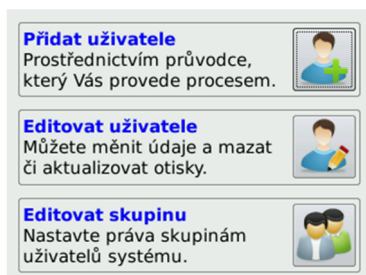
Obrázek 5–6: a) základní obrazovka b) modifikovaný odchod

Po přiložení prstu je na 2 sekundy zobrazeno jméno uživatele. Pomocí kláves 0 – 4 lze modifikovat odchod (Oběd, Pauza, Lékař, Dovolená, Služebně). Po stisku některé z těchto kláves se namísto hodin zobrazí uprostřed základní obrazovky indikátor typu progressbar, odpočítávající interval 5 sekund, v němž je nutno zaregistrovat modifikovaný odchod přiložením prstu. Úspěšnou operaci potvrdí název události společně se jménem a příjmením uživatele ve stavovém řádku viz *Obrázek 5–6b*.

Obdobným způsobem lze pomocí předvolby klávesou 6 vyvolat editační menu. Uživatel však musí mít oprávnění minimálně na úrovni skupinového správce.

## Editační menu

Ačkoliv je výrazně pohodlnější a rychlejší provádět změny v systému z WWW rozhraní, lze k tomuto účelu využít i GUI na LCD displeji. Podoba editačního menu je závislá na oprávnění uživatele, jimž bylo vyvoláno. Skupinový správce nemá možnost editovat skupiny, tudíž nemá tuto možnost v menu aktivní (je zobrazena šedě).



Obrázek 5–7: editační menu z pohledu hlavního správce

## Průvodce přidáním uživatele

Vložení nového uživatele probíhá formou průvodce na 4 obrazovkách, mezi nimiž lze přecházet navigačními tlačítky. První obrazovka slouží k zadání jména a příjmení pomocí kapacitní klávesnice, druhá stránka umožňuje načíst otisk prstu za účelem vytvoření jeho šablony. Nutné je sejmutí 2 otisků, přičemž uživatel je k přiložení/oddálení prstu vždy vyzván (popis procesu viz *Obrázek 5–4*). K první výzvě je za účelem demonstrace možností terminálu využit hlasový výstup v anglickém jazyce. Další obrazovka umožňuje zadání nepovinných identifikačních údajů – RFID karty a PINu. Prvky na poslední stránce jsou aktivní pouze v případě, kdy byl průvodce vyvolán hlavním správcem a slouží k volbě oprávnění a skupiny u nově přidávaného uživatele. Skupinový správce smí přidat běžného uživatele pouze do své skupiny.

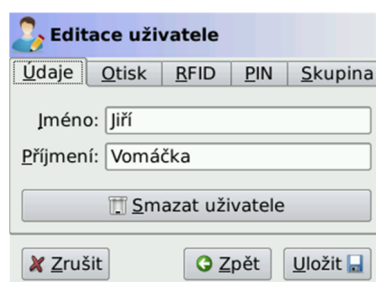


Obrázek 5–8: Průvodce přidáním uživatele - stránka s načtením otisku

V případě, kdy byl do systému přidán uživatel prostřednictvím webového rozhraní, a neexistuje tedy u něho v databázi šablona otisku, je po spuštění průvodce přednostně nabídnuta možnost načíst jeho otisk. Potvrzením lze přímo vyvolat obrazovku pro načtení otisku danému uživateli, stornování pak spustí kompletního průvodce od první stránky.

## Editace uživatele

V prvním kroku je zobrazena tabulka uživatelů v databázi (v případě skupinového správce pouze uživatelů v jeho skupině). Tabulka umožňuje vyhledávat na základě jednoho či více prvních písmen z příjmení osoby. Výběrem uživatele následuje přesun na obrazovku složenou z několika karet, umožňující změnu údajů, smazání osoby ze systému nebo aktualizaci otisku:



Obrázek 5–9: Obrazovka s editací uživatele

Tlačítko zpět umožňuje návrat k tabulce s výběrem uživatelů. Aktualizace otisku využívá k tomuto účelu modifikovanou stránku z průvodce přidáním uživatele. Obdobně jako při přidávání uživatelů nemůže skupinový správce měnit oprávnění ani příslušnosti ke skupině.

## Editace skupin

Zobrazena je tabulka s výběrem skupin, umožňující vybranou skupinu smazat nebo přejmenovat. Tlačítkem pod tabulkou lze vložit skupinu novou.



Obrázek 5–10: Editor skupin

## 5.5 Webové rozhraní

Kit MINI2440 obsahuje již v základu webový server boa (popsán v [4]). Ten je vhodný pro embedded aplikace díky minimálním nárokům na systémové prostředky. Jeho funkcionalita je však silně omezena a neodpovídá požadavkům na realizaci moderního rozhraní. Hlavní nedostatek pro tento projekt spočívá v absenci podpory skriptování na straně serveru v jazyku PHP. Proto byl zvolen jiný server – thttpd, který tuto možnost

nabízí a je např. oproti často používanému Apache stále realivně malý a nenáročný na zdroje (srovnávací tabulka viz [35]). Počáteční písmeno názvu programu je zároveň počátečním písmenem 3 slov shrnujících charakteristické vlastnosti – *turbo*, *tiny*, *throttling*. Značná nevýhoda tohoto serveru spočívá v chybějící podpoře zabezpečeného spojení – https, které je v současnosti na internetu již standardem. Z tohoto důvodu nelze doporučit přístup k webovému rozhraní zvenčí vnitrofiremní sítě. Pro účely vzdálené správy přes internet obsahuje terminál jiný kanál, který je již šifrován (viz kapitola 5.6.3).

Pro úspěšné spuštění `thttpd` s podporou PHP na kitu MINI2440 byl využit postup popsáný v [34]. V prvním kroku je třeba zkompilevat PHP a nakonfigurovat ho pro využití s `thttpd` (php nutno přeložit s volbou `--with-thttpd`, následovanou cestou ke zdrojovým kódům webserveru). V kompilaci PHP byla za účelem zachování co nejmenší velikosti výsledného souboru vypnuta volbou `--disable-all` všechna rozšíření. Opět aktivována pak byla pouze ta nezbytná po správný chod vytvořených webových stránek:

```
--with-sqlite3
--with-session
--enable-hash
```

Po kompilaci `php` s výše uvedenými parametry zbývá zkompilevat samotný webový server. Výsledná velikost spustitelného souboru je cca 11 MB a je automaticky spuštěn při startu systému. Výchozí adresář pro soubory související s webovým rozhráním má cestu `/www`.

### 5.5.1 JQuery

Základ webové aplikace je vystavěn na upraveném prostředí JQueryUI [36], které, stejně jako samotné `www` stránky, využívá javascriptovou knihovnu JQuery [37]. Jedná se o knihovnu, umožňující provádět relativně jednoduše široké spektrum operací na straně klienta – ve webovém prohlížeči. Prostřednictvím svého API umožňuje vytvářet interaktivní `html` prvky i celé dynamické webové aplikace. Jejím jádrem je algoritmus nazvaný Sizzle [37], realizující výběr konkrétního `html` prvku (objektu) na stránce. Takto vybranému prvku lze pak např. měnit vlastnosti nebo ho ze stránky odebrat. Další funkcí, poskytovanou JQuery a využitou v tomto projektu je asynchronní komunikace se serverem bez nutnosti obnovit celou stránku. Tento způsob komunikace bývá označován jako Ajax.

Knihovny JQuery i JQueryUI jsou načítány z externího zdroje na internetu. Výhodou je menší zatížení serveru v MINI2440 nebo rychlejší načtení aplikace v případě, kdy se knihovna JQuery ze stejného zdroje ve vyrovnávací paměti prohlížeče již nachází. Využita je CDN (*Content delivery network*) společnosti Google [38]. Pro případ kdy se nelze připojit k internetu je k dispozici i lokální kopie knihoven v MINI2440 (tzv. *fallback*

řešení). Knihovny se načítají pouze jednou, a to po přihlášení uživatele do webové aplikace.

### 5.5.2 Struktura

Webové rozhraní obsahuje několik modulů, reprezentovaných php skripty. Dále pak nezbytné pomocné soubory s javascriptem (přípona .js) a kaskádovými styly (přípona .css). Hlavní moduly a jejich funkce si nyní uvedeme:

- **Index.php** – hlavní stránka, obsahuje lištu s menu, aktuálním datem a přihlášeným uživatelem a dále též záhlaví a zápatí stránky. Moduly vyvolané z menu jsou načítány pomocí Ajaxu.
- **Attend.php** – realizuje výpis docházky a umožňuje do určité míry její editaci.
- **Users.php** – poskytuje výpis uživatelů. Lze je editovat, přidávat i mazat.
- **Groups.php** – umožňuje základní operace se skupinami
- **Stats.php** – zobrazuje odpracovanou dobu za jednotlivé dny a celkovou bilanci fondu pracovní doby – tzv. saldo
- **Settings.php** – umožňuje změnit heslo přihlášenému uživateli
- **Login.php, logout.php, authenticate.php** – zajišťují funkce svázané s přihlašováním do systému

Obrázkové ikony webového rozhraní byly převzaty z internetové banky ikon [39] a jsou volně šiřitelné pro nekomerční účely.

### 5.5.3 Použití webového rozhraní

Po zadání adresy serveru (kitu MINI2440) do prohlížeče proběhne přesměrování na přihlašovací stránku login.php. Pro přihlášení je třeba zadat příjmení, jméno a heslo. Přenos hesla probíhá v šifrované podobě, kdy je využita hashovací funkce sha-1. Pro zvýšení odolnosti vůči odposlechu se k heslu před hashováním přidá aktuální časové razítko. Tento postup se nazývá zasolení a značně ztěžuje útočníkovi útok pomocí slovníku a přihlášení pomocí odposlechnutých dat. Hash hesla zasolen časovým razítkem má totiž platnost 30 sekund, stejně jako přihlašovací formulář.

Po úspěšném přihlášení je uživatel přesměrován do hlavního rozhraní, jehož podoba se odvíjí od uživatelského oprávnění. Skupinový správce nemá k dispozici kartu s editací skupin a běžný uživatel pak ani kartu s výpisem uživatelů. V následujícím popisu vždy předpokládáme, že se jedná o hlavního správce.

## Docházka

Ve všech případech je po přihlášení zobrazena karta z docházkou a předvyplněným vlastním jménem v poli filtrace záznamů. Stiskem tlačítka filtrovat lze tak ihned zobrazit vlastní docházku. Toto pole umožňuje rychlé zadávání jmen pomocí tzv. našeptávače, který využívají například internetové vyhledávače (viz *Obrázek 5–11*). U skupinového správce jsou nabízená jména omezena jen na jeho skupinu, běžný uživatel pak může zobrazit jen svou docházku. Pokud je pole prázdné, zobrazuje se docházka všech uživatelů (možné pouze u hlavního správce).

Jméno	Příjmení	Typ	Datum	Editace
Vilemína	Konečná	Příchod	2013-05-12 21:03:05	
Vilemína	Konečná	Odchod - oběd	2013-05-12 21:03:12	
Vilemína	Konečná	Příchod	2013-05-12 22:20:31	
Miloš	Petrovský	Příchod	2013-05-12 22:20:40	
Miloš	Petrovský	Odchod	2013-05-12 22:21:05	
Věromír	Nejezchleba	Příchod	2013-05-12 22:21:17	
Zdeněk	Pohlreich	Odchod - služebně	2013-04-01 14:04:19	N/A
Miloš	Petrovský	Odchod - lékař	2013-04-05 11:11:11	N/A
Vilemína	Konečná	Příchod	2013-04-08 22:49:04	N/A
Marcela	Mrkos	Odchod	2013-04-08 22:53:24	N/A
Marcela	Mrkos	Odchod - oběd	2013-04-08 22:53:37	N/A
Ahmed	Eba	Odchod - pauza	2013-04-08 23:04:29	N/A
Ahmed	Eba	Odchod - lékař	2013-04-08 23:08:38	N/A
Ahmed	Eba	Odchod	2013-04-08 23:52:46	N/A

Díplomová práce - Martin Tříška @ FEKT VUT Brno - 2013

Obrázek 5–11: Výpis docházky a ukázka našeptávače

Filtrovat lze kromě jména i na základě typu události (příchod/odchod, varianty odchodů) a data, kdy lze využít rychlé volby a zobrazit docházku za dnešní den, aktuální týden nebo měsíc. Po kliknutí na řádek *Od* nebo *Do* se zobrazí kalendář pro snadný výběr data. U záznamů za aktuální den je možná i jejich editace. Měnit lze typ, datum a čas, nebo je možno záznam smazat. Změny smí provádět pouze správce.



## Uživatelé

Karta Uživatelé obsahuje seznam uživatelů v obdobném formátování, jaké používá výpis docházky. Z postranního panelu lze uživatele přidat, kliknutím na symbol tužky ve sloupci editace pak také editovat. Skupinový správce má přístup pouze ke své skupině. Editace probíhá z „vyskakovacího“ formuláře (stejně jako na jiných kartách).

Obrázek 5–12 : Editace uživatele

Uživateli, vloženému přes webové rozhraní, je nutno načíst otisk pomocí GUI na LCD displeji (viz kapitola 5.4.3). Po vyvolání průvodce přidáním uživatele je tato možnost nabídnuta přednostně a v případě, kdy databáze obsahuje uživatelů bez otisku více, je možno z nich vybírat.

## Skupiny

Tato karta obsahuje výpis skupin a umožňuje jejich přidání, přejmenování nebo smazání. Tabulka s výpisem obsahuje i sloupec s přístupovými právy, který je připraven pro případ provázání docházkového systému se systémem přístupovým. Struktura seznamu se podobá tabulce uživatelů.

## Statistiky

Karta formou tabulky zobrazuje odpracovanou dobu za jednotlivé dny od začátku měsíce. Za každý den je také zobrazen čas strávený na obědě a ostatní přerušení pracovní doby (např. kuřácká pauza). Součet všech těchto údajů je uveden jako celková doba pobytu v práci. Celková bilance pracovní doby za daný měsíc – saldo, se nachází v postranním sloupci.

## Nastavení

Jednoduchý formulář na této kartě umožňuje změnu hesla přihlášeného uživatele. Všechny hesla jsou ukládána do databáze v hashované podobě.

## 5.6 Ostatní části SW výbavy

V řídicím systému tvořeném kitem MINI2440 jsou krom hlavních softwarových celků, popsaných v předchozích podkapitolách, využity i další programy a skripty. Ty zajišťují především podpůrné funkce. Některé z nich byly vytvořeny přímo pro účely tohoto projektu, jiné byly přeloženy z převzatých open-source zdrojových kódů (jejich souhrn je uveden na konci této podkapitoly).

### 5.6.1 Statsgen

Je program, sloužící ke generování údajů o odpracované době jednotlivých uživatelů, které ukládá do tabulky evidence v hlavní databázi. Spouští se denně prostřednictvím skriptu *daily.sh*. Údaje jsou generovány na základě denní databáze *daily.db* vždy zpětně za uzavřený den. Tabulka evidence má strukturu:

User_id	Saldo	1	2	...	31
---------	-------	---	---	-----	----

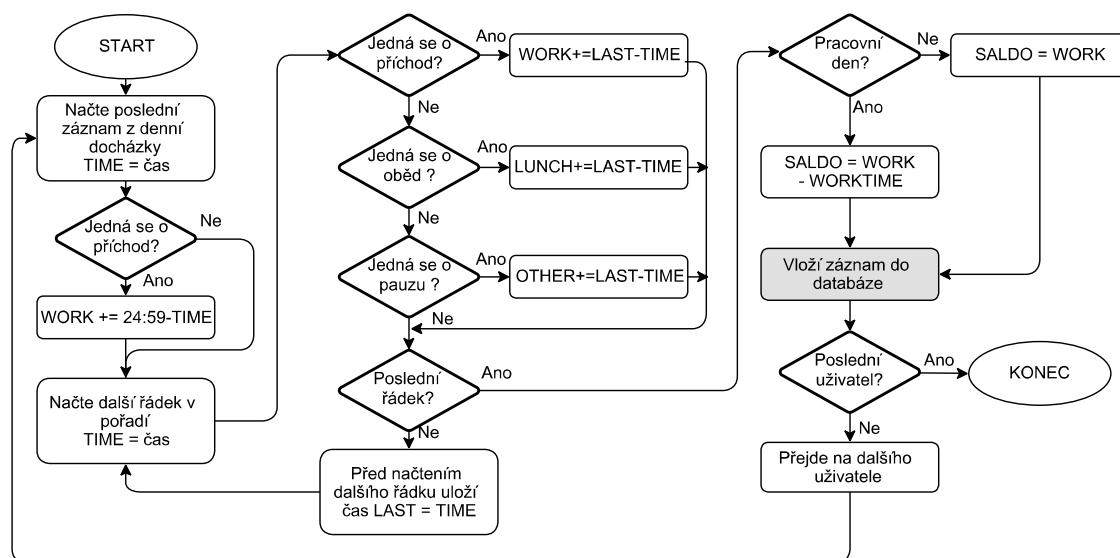
Tabulka 17: Struktura tabulky evidence

Tvar denního záznamu (ve sloupcích 1 – 31, dle dne v měsíci) má tvar:

ODPRACOVANÁ\_DOBA; OBĚD; OSTATNÍ\_PŘERUŠENÍ

Např.: 7:35; 0:27; 0:12

Program počítá jednotlivé údaje podle algoritmu:



Obrázek 5–13: Zjednodušený postup výpočtu pracovní doby

Hodnota saldo je zaznamenávána ve vteřinách a odpovídá odpracované době, snížené o stanovenou denní pracovní dobu. Může tak nabývat i záporného znaménka. To však neplatí v případě sobot, nedělí a svátků (definovaných v tabulce *svatky*), kdy se odpracovaná doba nesnižuje. Takto získaný údaj je následně přičten k hodnotě ve sloupci *Saldo*.

## 5.6.2 Skripty

Některé z pomocných funkcí jsou realizovány skripty pro příkazový interpret, spouštěnými při určitých událostech. Nejdůležitější z nich je *etc/init.d/rcS*, volaný automaticky po startu OS provádí tyto operace:

- Systémová nastavení, týkající se běhového režimu OS, souborového systému, proměnných prostředí apod.
- Spustí logovacího démona *syslogd* a démona časovaného spouštění *crond*
- Zobrazí na displeji úvodní logo
- Spustí základního síťového démona *inetd*, nastaví síť a nastartuje webový server *thttpd*
- Obnoví systémový čas z interního *rtc* a spustí jeho průběžnou synchronizaci pomocí démona *ntpd*
- Definuje proměnné prostředí, potřebné pro běh Qt aplikace (cesty ke knihovnám, rozlišení a orientace displeje, vstupní zařízení – touch panel).
- Připraví databázi *daily.db* v paměti RAM (pomocí skriptu *dailyprepare.sh*)
- Spustí na pozadí program *prstd* (který sám spustí proces vizualizace)
- Přehraje úvodní znělku

Další skripty – *daily.sh* a *monthly.sh* jsou automaticky volány vždy při přelomu dne a měsíce prostřednictvím služby *crond*. Ze skriptu *daily.sh* je spouštěn program *statsgen*, je proveden převod údajů o docházce z denní do hlavní databáze a ta se prostřednictvím protokolu *ftp* zálohuje na externí server. Skript *monthly.sh* přesouvá obsah tabulek *evidence* a *dochazka* do tabulek *evidence\_last* a *dochazka\_last*. Zdrojové tabulky pak připraví pro další měsíc (smaže údaje o docházce a vynuluje odpracovanou dobu).

Skript *prstflash.sh* slouží pro aktualizaci firmware v MCU rozšiřující desky. Jedná se o wrapper programu *stm32flash*, popsaného dále. Tento skript očekává jako parametr cestu k souboru s firmwarem ve formátu Intel HEX. Nejprve zašle prostřednictvím hlavní aplikace *prstd* příkaz k restartu desky s následným skokem do bootloaderu, poté tuto aplikaci stejně jako proces vizualizace ukončí. Následuje samotný download firmwaru po sériové lince programem *stm32flash*, zakončený skokem na adresu *0x00*, odkud se nově nahraný program začne vykonávat. Posledním krokem je opětovné spuštění hlavního

procesu prstů a zápis do logu. Po celou dobu běhu se na displeji zobrazuje upozornění na dočasnou nefunkčnost terminálu.

### 5.6.3 Převzaté programy

Součástí SW vybavy tvoří i programy, které nebyly vytvořeny autorem práce a nebyly v této práci dosud popsány. Jsou obvykle přeloženy ze zdrojových kódů pomocí překladače GCC, nebo je kit MINI2440 obsahuje již z výroby. Uvedme si nyní jejich stručný přehled:

- **NTP** – jedná se balíček programů, zajišťující služby spojené se synchronizací času. Obsahuje utilitu `ntpd`, zajišťující jednorázovou synchronizaci a především démona `ntpd`, sloužícího k průběžné korekci času pomocí fázového závěsu. Jako zdroj synchronizačních dat je využit tzv. *pool* (jedenkrát za hodinu měněný výběr) serverů z ČR s adresou `cz.pool.ntp.org`. Balíček zdrojových kódů je ke stažení na [40].
- **OpenSSL** – je balíček utilit, týkajících se šifrované komunikace. V této aplikaci je využit démon `sshd`, nahrazující nešifrovaný komunikační kanál - telnet. Stránky projektu Open SSH pro MINI2440 jsou dostupné z [41].
- **Stm32flash** [42] – program pro příkazový interpret, umožňující programování MCU výrobce STM pomocí vestavěného bootloaderu. Bylo nutné ho upravit pro podporu pro MCU s flash pamětí větší než 128 kB (označované výrobcem jako high-density).
- **Madplay** – mp3 přehrávač pro příkazový interpret (je v kitu MINI2440 již z výroby)
- **Busybox** – sada standardních linuxových příkazů, realizovaných jedním spustitelným souborem, který je volán pomocí odkazů s názvy příkazů. Tento princip se označuje jako multi-call binary a využíván je často v embedded zařízeních kvůli eliminaci nutnosti velkého množství malých jednoúčelových programů. Na kitu je ve verzi 1.13.3 již z výroby.

### 5.7 Spotřeba energie

Ačkoliv je energetická náročnost zařízení veličinou spadající pod hardware terminálu, závisí také na softwaru. To dokazuje *Tabulka 18*. Proto nemohla být spotřeba zjištěna před samotným vývojem SW vybavy.

<b>Režim</b>	<b>Proudový odběr [mA]</b>	<b>Příkon [W]</b>
<b>Aktivní, 100% vytížení CPU</b>	847 mA	4,24
<b>Aktivní, běžící základní aplikace</b>	710 mA	3,55
<b>Úsporný režim</b>	503 mA	2,51

*Tabulka 18 : Energetická náročnost terminálu*

Jedním ze způsobů jak snížit náklady na provoz zařízení je minimalizace spotřeby elektrické energie. Proudové požadavky terminálu, vyvinutého v této práci udává *Tabulka 18*. Další prostor ke snižování energetických nároků poskytuje výměna kitu MINI2440 za hospodárnější platformu, například CPU s jádrem ARM Cortex. K dalšímu mírnému snížení spotřeby by přispělo vypínání RFID čtečky v úsporném režimu.

## 6 ZÁVĚR

V této diplomové práci byl vyroben hardware terminálu docházkového systému podporující identifikaci uživatele pomocí 3 základních faktorů znalostí – PIN kódem, předmětem – RFID kartou a biometrickým údajem ve formě otisku prstu. Pro tento terminál byla dále vyvinuta ukázková programová vybava, která demonstruje jeho nejdůležitější funkce. Během vývoje byl kladen důraz na využití moderních komponent, příjemné uživatelské rozhraní, další rozšiřitelnost a v neposlední řadě i na cenu.

Práce volně navazuje na autorovo předchozí dílo, které se zabývalo vývojem jednoduchého terminálu pro přístupový systém včetně komunikace s nadřazeným systémem – serverem. Propojením těchto dvou produktů tak vznikne komplexní celek, řešící jak docházku, tak řízení přístupů, jehož server slouží zároveň i jako hlavní docházkový terminál. Svým provedením je předurčen především pro malé firmy.

Hlavní hardwarovou částí je vývojový kit, osazený výkonným procesorem ARM, doplněný o rozšiřující desku. Ta využívá vlastní MCU s jádrem ARM Cortex M3 a vykonává některé nízko úrovněvé operace, které tak zbytečně nezatěžují hlavní platformu. Rozšiřující deska je koncipována univerzálně a umožňuje využití i s jiným nadřazeným systémem v jiné aplikaci. Terminál je dále vybaven záložním zdrojem, zvukovým výstupem, kapacitní klávesnicí a některými dalšími hardwarovými celky. Po mechanické stránce byl veškerý HW instalován do hliníkového pouzdra, uzpůsobeného potřebám aplikace CNC obráběním. Po softwarové stránce byla v této práci vytvořena a popsána ukázková aplikace skládající se z několika celků a demonstrující hlavní funkce terminálu. Vzhledem k faktu, že možnosti nabízené zvolenou HW platformou jsou velmi široké, ji nelze chápat jako definitivní verzi, určenou pro koncového zákazníka.

Možnosti rozšíření SW vybavy jsou značné. V GUI by bylo například vhodné umožnit uživateli zobrazit při odchodu stručný náhled na odpracovanou dobu, případně ho hlasovým výstupem pozdravit. Dále lze díky současným webovým technologiím na stránky přidat například grafické znázornění statistik, přímý export pracovních výkazů do souboru pdf a spoustu dalších užitečných prvků. Cílem této práce však nebylo vytvořit maximálně komplexní řešení, ale spíše ukázat jednu z cest, po které lze dojít ke konkrétní funkcionalitě, požadované uživatelem, potažmo zákazníkem. Toto se, myslím, podařilo splnit.

# Literatura

- [1] INNOVATRICES, Fingera™ - Webová prezentace systému [ONLINE]. Fingera s.r.o. 2013. [cit. 2013-4-20]. Dostupné z WWW: <<http://www.fingera.com/cs/>>
- [2] ACS Line, *Elektronický identifikační systém* [ONLINE]. ESTELAR s.r.o. 2013. [cit. 2013-4-21]. Dostupné z WWW: <<http://www.acsline.cz/cs/>>
- [3] MIAXIS, *Fingerprint Access Control* [ONLINE]. Miaxis Biometrics Co., Ltd 2011. [cit. 2013-4-21]. Dostupné z WWW: <<http://www.acsline.cz/cs/kontakty>>
- [4] TRÍSKA, M. *Přístupový systém se snímači otisků prstů*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2011. 56 s. Vedoucí bakalářské práce Ing. Tomáš Macho, Ph.D.
- [5] FRIENDLY ARM, *Mini 2440 Hardware overview*. Guangzhou FriendlyARM Computer Technology. 2010, 12s. Dostupné z WWW: <[http://www.friendlyarm.net/dl.php?file=mini2440\\_overview.pdf](http://www.friendlyarm.net/dl.php?file=mini2440_overview.pdf)>
- [6] MICROCHIP, *MCP1701AT Data Sheet*. Microchip Technology Inc. 2007, 22s. [cit. 2012-10-11]. Dostupné z WWW: <<http://ww1.microchip.com/downloads/en/DeviceDoc/21991c.pdf>>
- [7] TRACO, *TSR-1 Series datasheet*, Tracopower, [cit. 2012-10-11]. Dostupné z WWW: <[http://www.tracopower.com/fileadmin/medien/dokumente/pdf/product\\_selection/TSR1.pdf](http://www.tracopower.com/fileadmin/medien/dokumente/pdf/product_selection/TSR1.pdf)>
- [8] STM, *STM32F100RC Data Sheet*. STMicroelectronic, September 2012, 98s. [cit. 2012-04-24]. Dostupné z WWW: <<http://www.st.com/st-web-ui/static/active/en/resource/technical/document/datasheet/CD00212417.pdf>>

- [9] STM, *AN2586 Application note Getting started with STM32F10xxx hardware development*. STMicroelectronic November 2011, 28s. [cit. 2012-04-24].  
Dostupné z  
WWW:<[http://www.st.com/internet/com/TECHNICAL\\_RESOURCES/TECHNICAL\\_LITERATURE/APPLICATION\\_NOTE/CD00164185.pdf](http://www.st.com/internet/com/TECHNICAL_RESOURCES/TECHNICAL_LITERATURE/APPLICATION_NOTE/CD00164185.pdf)>
- [10] STM, *AN2867 Application note Oscillator design guide for STM microcontrollers*. STMicroelectronic March 2011, 24s. [cit. 2012-04-24]. Dostupné z WWW:  
<[http://www.st.com/internet/com/TECHNICAL\\_RESOURCES/TECHNICAL\\_LITERATURE/APPLICATION\\_NOTE/CD00221665.pdf](http://www.st.com/internet/com/TECHNICAL_RESOURCES/TECHNICAL_LITERATURE/APPLICATION_NOTE/CD00221665.pdf)>
- [11] Fonter electronics, *Quartz crystal resonator datasheet*. Fronter electronics Co., Ltd, 1s. [cit.2012-04-24]. Dostupné z WWW: <  
<http://www.gme.cz/dokumentace/131/131-059/dsh.131-059.1.pdf>>
- [12] JYE, F.C.: *Calculate the LED Lifetime Performance in Optocouplers to Predict Reliability*. AVAGO Technologies February 2012, 5s. [cit. 2013-02-14]. Dostupné z WWW: <<http://www.avagotech.com/docs/AV02-3401EN>>
- [13] COSMO, *KP1020 Datasheet*. Cosmo electronics corporation. 2005, 2s. [cit. 2013-02-14]. Dostupné z WWW: <<http://www.gme.cz/dokumentace/523/523-055/dsh.523-055.1.pdf>>
- [14] FORWARD RELAYS, *P-5 Datasheet*. Ningbo Forward Relay Corporation LTD. 2s. [cit. 2013-02-14]. Dostupné z WWW: <  
<http://www.gme.cz/dokumentace/634/634-190/dsh.634-190.1.pdf>>
- [15] TI, *TPA2006D1 Datasheet*. Texas Instruments, September 2006, 23s. [cit. 2012-04-24]. Dostupné z WWW: < <http://www.farnell.com/datasheets/1383803.pdf> >



- [16] Microchip, *MCP73837 Datasheet*. Microchip Technology Inc. 2011, 34s. [cit. 2012-04-24]. Dostupné z WWW: < <http://www.farnell.com/datasheets/1538062.pdf> >
- [17] TI, *TPS61252 Datasheet*. Texas Instruments, September 2010, 23s. [cit. 2012-04-24]. Dostupné z WWW: < <http://www.farnell.com/datasheets/1362620.pdf> >
- [18] ON, *MBRA210LT3G Datasheet*. ON Semiconductor, January 2012, 5s. [cit. 2012-04-29]. Dostupné z WWW: < <http://www.farnell.com/datasheets/1362620.pdf> >
- [19] microBuilder.eu, *Measuring Battery Voltage with an ADC, 2010*, [cit. 2012-04-29] Dostupné z WWW: <<http://www.microbuilder.eu/Tutorials/Fundamentals/MeasuringBatteryVoltage.aspx>>
- [20] FREESCALE, *MPR121QR2 Datasheet*. Freescale Semiconductor, February 2013, 27s. [cit. 2013-04-20]. Dostupné z WWW: <[http://www.freescale.com/files/sensors/doc/data\\_sheet/MPR121.pdf](http://www.freescale.com/files/sensors/doc/data_sheet/MPR121.pdf)>
- [21] FLAJZAR, *Sortiment nabídky Flajzar* [on-line]. [cit. 2012-04-29]. Dostupné z WWW: <<http://flajzar.cz/>>
- [22] SEMACH, *Technologické povrchy plošných spojů*. Semach - plošné spoje [cit. 2012-11-25] . Dostupné z WWW: <<http://semach.cz/povrchy.html>>
- [23] FREESCALE, *AN3747 - MPR121QR2 Pad layout*. Freescale Semiconductor, October 2009, 5s. [cit. 2012-18-11]. Dostupné z WWW: <[http://www.freescale.com/files/sensors/doc/app\\_note/AN3747.pdf](http://www.freescale.com/files/sensors/doc/app_note/AN3747.pdf)>

- [24] MIAxis, *SM-621 Fingerprint Verification Module User Manual*. Maxis Biometrics Co., Ltd. 2007, 36s. [cit. 2010-10-11]. Dostupné z WWW: <<http://www.olimex.cl/pdf/Sensors/SM621%20FINGERPRINT%20VERIFICATION%20MODULE%20USER%20MANUAL.pdf>>
- [25] ELING, *Sortiment společnosti* [ONLINE]. Eling Bohemia s.r.o. 2012, [cit. 2012-04-24]. Dostupné z WWW: < <http://www.eling.cz/> >
- [26] RAYFILM, *Sortiment společnosti* [ONLINE]. RAYFILM s.r.o. [cit. 2012-04-24]. Dostupné z WWW: < <http://www.rayfilm.cz/> >
- [27] COOCOX, *CoIDE* [cit. 2012-04-24]. Dostupné z WWW: <[http://www.coocox.org/CooCox\\_CoIDE.htm](http://www.coocox.org/CooCox_CoIDE.htm)>
- [28] STM. Datasheet STM32VLDISCOVERY User manual [online]. [cit. 2012-04-27]. Dostupné z WWW: <[http://www.st.com/internet/com/TECHNICAL\\_RESOURCES/TECHNICAL\\_LITERATURE/USER\\_MANUAL/CD00267113.pdf](http://www.st.com/internet/com/TECHNICAL_RESOURCES/TECHNICAL_LITERATURE/USER_MANUAL/CD00267113.pdf)>.
- [29] FREESCALE, *AN3944 - MPR121QR2 Quick start guide*. Freescale Semiconductor, March 2010, 5s. [cit. 2012-18-11]. Dostupné z WWW: <[http://www.freescale.com/files/sensors/doc/app\\_note/AN3944.pdf](http://www.freescale.com/files/sensors/doc/app_note/AN3944.pdf)>
- [30] Nokia, *Qt Project* [online]. [cit. 2012-4-27] Dostupné z WWW: < <http://www.developer.nokia.com/Develop/Qt/>>
- [31] DIGIA, *Qt Project* [online]. [cit. 2012-4-27] Dostupné z WWW: <<http://qt-project.org/>>
- [32] SQLite, *About SQLite; SQLite Download Page* [online]. [cit. 2012-11-15] <<http://www.sqlite.org/>>

- [33] SERENO-LABS, *Qt 4.6.2 installation procedure for FriendlyARM mini 2440 Linux*, Jun 2010 [cit. 2012-08-29] Dostupné z WWW:   
<<http://www.sereno-labs.com/qt-4-6-2-installation-procedure-for-friendlyarm-mini-2440-board>>
- [34] FESTING D.: *thttpd-2.21b + php-5.3.6*, July 2011 [cit. 2012-11-19] Dostupné z WWW: <<http://www.friendlyarm.net/forum/attachment/19726>>
- [35] ACME, *thttpd - tiny/turbo/throttling HTTP server; Web Server Comparisons*, [cit. 2012-11-19] Dostupné z WWW: <<http://acme.com/software/thttpd/>>
- [36] JQUERY, *User interface* , The jQuery Foundation, [cit. 2013-04-29] Dostupné z WWW: <<http://jqueryui.com/>>
- [37] JQUERY, *Javascript library* , The jQuery Foundation, [cit. 2013-04-28] Dostupné z WWW: <<http://jquery.com/>>
- [38] GOOGLE, *CDN Hosted libraries*. [cit. 2013-04-28] Dostupné z WWW: <<https://developers.google.com/speed/libraries/>>
- [39] ICONSPEDIA, *Your source of free icon*. [cit. 2013-04-29] Dostupné z WWW: <<http://www.iconspedia.com/>>
- [40] NTP, *The Network Time Protocol*. [cit. 2013-04-29] Dostupné z WWW: <<http://www.ntp.org/>>
- [41] OPENSSSH4MINI2440, *Porting the openssh to mini2440*. [cit. 2013-04-29] Dostupné z WWW: < <https://code.google.com/p/openssh4mini2440/>>
- [42] STM32FLASH, *Open source cross platform flash program for the STM32 bootloader*. [cit. 2013-02-11] Dostupné z WWW: < <https://code.google.com/p/stm32flash/>>

- [43] MARGORÍN, M. *jQuery bez předchozích znalostí*. Praha: COMPUTER PRESS, 9.2011. 256 s. ISBN 978-80-251-3379-8
  
- [44] HOGAN, B. *HTML5 a CSS3 – Výukový kurz webového vývoje*. Brno: COMPUTER PRESS, 2012. 272 s. ISBN 978-80-251-3576-1

# Seznam příloh

Příloha A. Seznam součástek

Příloha B. Schéma rozšiřující desky

Příloha C. Deska plošných spojů

Příloha D. Osazovací výkresy

Příloha E. Deska plošných spojů kapacitní klávesnice

Příloha F. Výkres čelního panelu

Příloha G. Potisk čelního panelu

Příloha H. Fotografie realizovaného přístroje a osazené desky

Příloha: CD se zdrojovými kódy

Příloha A – Seznam součástek

OZNAČENÍ	HODNOTA	POUZDRO
----------	---------	---------

**REZISTORY**

<b>R1,R8,R9,R30,R31,R32,R35</b>	10k	0805
<b>R2,R44</b>	0R	0805
<b>R3,R4</b>	10R	0805
<b>R5,R7, R13,R14, R15,R16</b>	390R	0805
<b>R6</b>	220R	0805
<b>R10,R11</b>	4k7	0805
<b>R12</b>	75k 1%	0805
<b>R17,R18,19,R20</b>	1k5	0805
<b>R21</b>	50k	0805
<b>R22,R23,R24,R25,R29,R34</b>	100k	0805
<b>R26,R37,R38,R39,R40,R41,R42,R43</b>	1k	0805
<b>R27</b>	240k	0805
<b>R28</b>	750k	0805
<b>R33</b>	2k	0805
<b>R36</b>	22k	0805

**KAPACITORY**

<b>C1,C22,C25</b>	4u7	SMA
<b>C2,C3,C4,C5,C6,C7,C8,C9,C13,C16,C17</b>	100n	0805
<b>C3</b>	470u	CLH1010
<b>C10</b>	10u	SMB
<b>C11,C12,C14,C15,C26</b>	8n2	0805
<b>C20,C23</b>	100p	0805
<b>C21</b>	1u	0805
<b>C24</b>	22u	SMC
<b>C27,C28</b>	30p	0805

**INDUKČNOSTI**

<b>L1</b>	1u	1812
<b>FB2,FB3,FB4</b>	Ferrite bead 2 A	1206
<b>FB5</b>	Ferrite bead 500 mA	1206
<b>OZNAČENÍ</b>	<b>HODNOTA / TYP</b>	<b>POUZDRO</b>

## POLOVODIČE

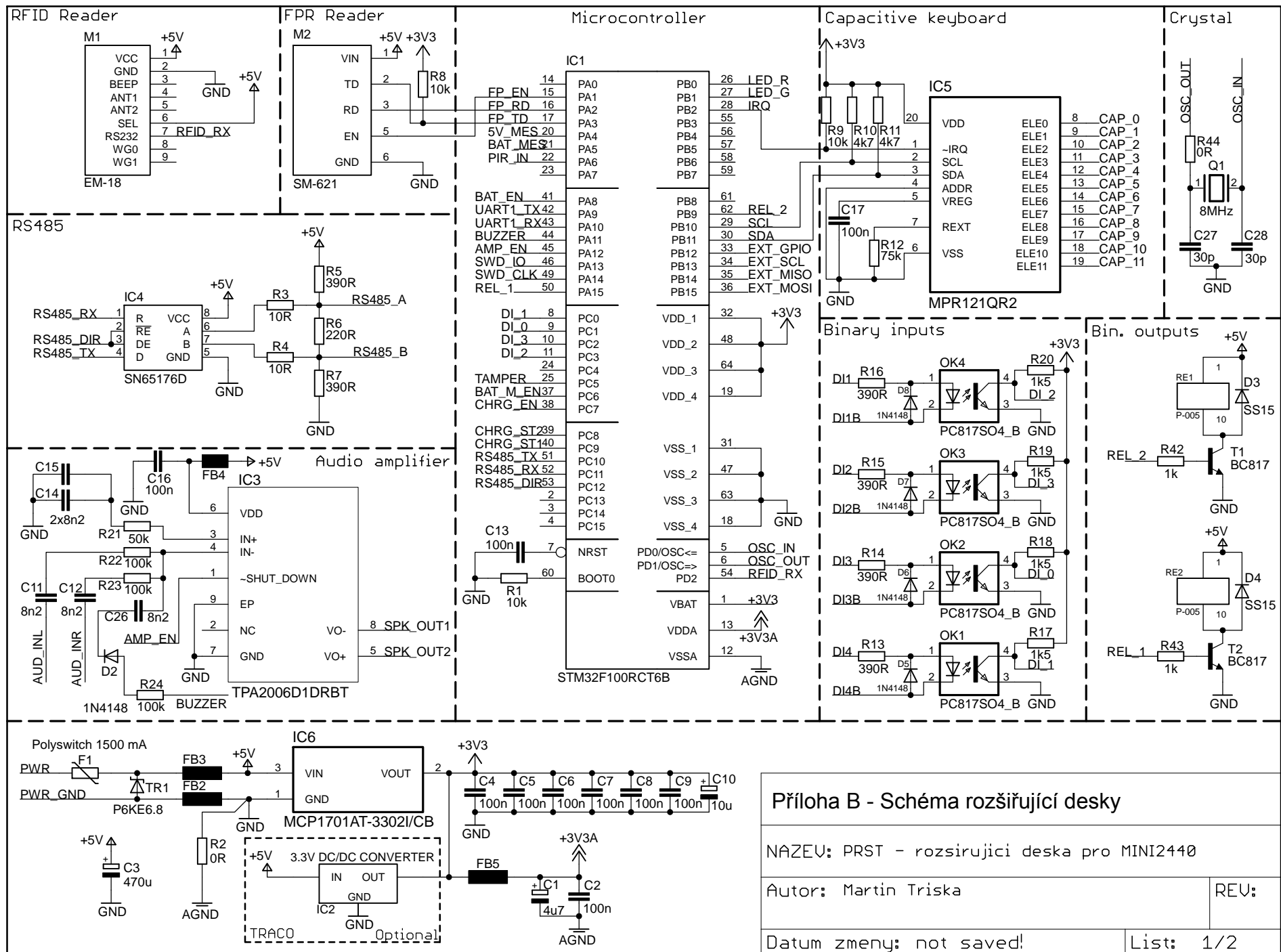
<b>IC1</b>	STM32F100RCT6B	TQFP64
<b>IC2</b>	Traco TSR1-2433	
<b>IC3</b>	TPA2006D1	8SON
<b>IC4</b>	SN65176	SO-08
<b>IC5</b>	MPR121QR2	QFN20
<b>IC6</b>	MCP1701AT	SOT95
<b>IC7</b>	TPS61252	QFN8
<b>IC8</b>	MCP73837-FCI/UN	MSOP10
<b>D2, D5, D6, D7, D8</b>	1N4148	SOD123
<b>D3, D4</b>	SS15	SMB
<b>D9</b>	MBRA210LT3G	SMB
<b>OK1,OK2,OK3,OK4</b>	PC817	SO-4
<b>T1,T2,T3</b>	BC817	SOT23
<b>T4</b>	BSS84	SOT23
<b>TR1</b>	P6KE6.8	SMB
<b>TR2,TR3</b>	SM6T6V8CA	SMB
<b>F1</b>	Polyswitch 1,5 A	1812

## KONEKTORY

<b>K1,K5</b>	15EDGRC-3.5/2P
<b>K2</b>	JST-PH3
<b>K3</b>	PSH-03P
<b>K4,K9,K10,K11</b>	JST-PH2
<b>K6</b>	15EDGRC-3.5/8P
<b>K7,K8</b>	15EDGRC-3.5/3P
<b>K12</b>	JST-PH13

<b>K13.K14,K15</b>	JST-PH3
<b>K16,SV1,SV2</b>	2,54 mm pin header
<b>OSTATNÍ</b>	
<b>M1</b>	RFID READER EM-18
<b>M2</b>	Fingerprint module SM-621
<b>Q1</b>	8MHz crystal                      HC49-S
<b>RE1,RE2</b>	RELEMP-05





## Příloha B - Schéma rozšiřující desky

NAZEV: PRST - rozširujici deska pro MINI2440

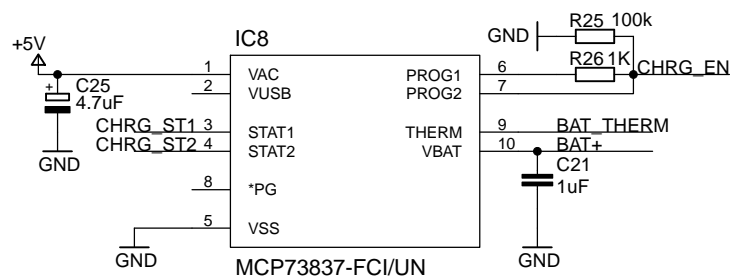
Autor: Martin Triska

REV:

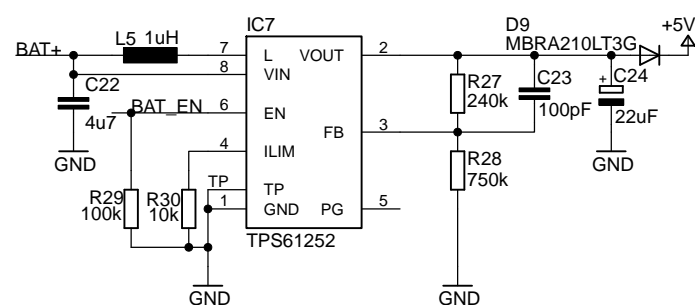
Datum zmeny: not saved!

List: 1/2

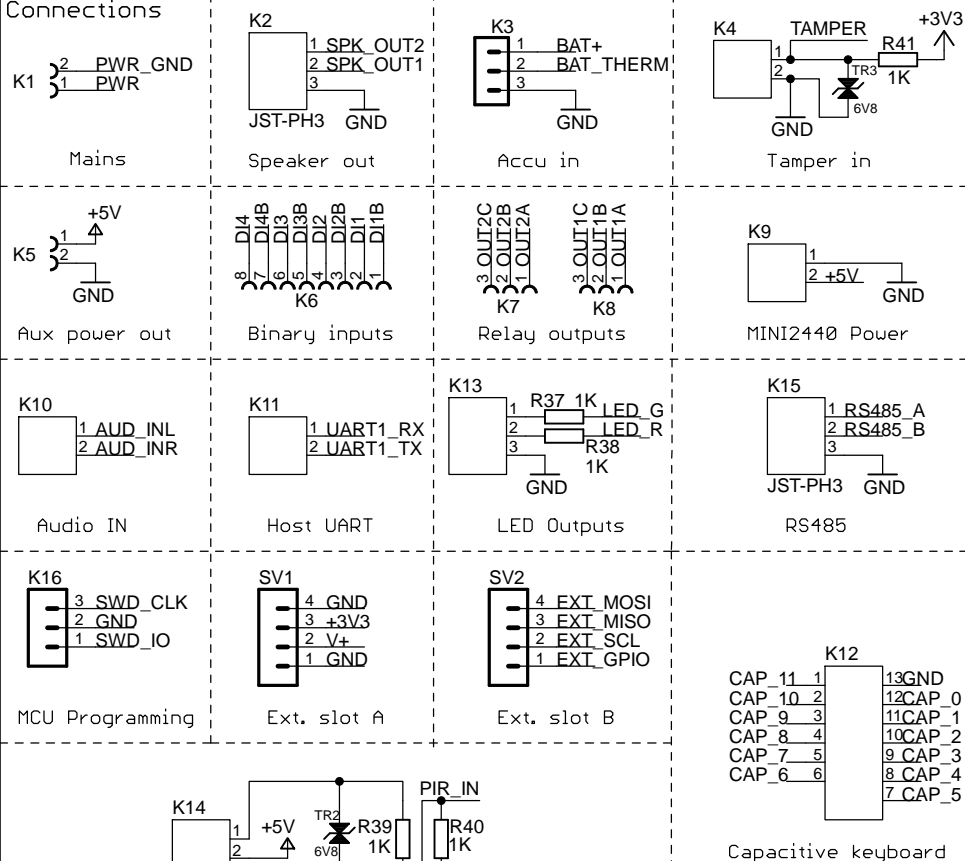
## Accu charger



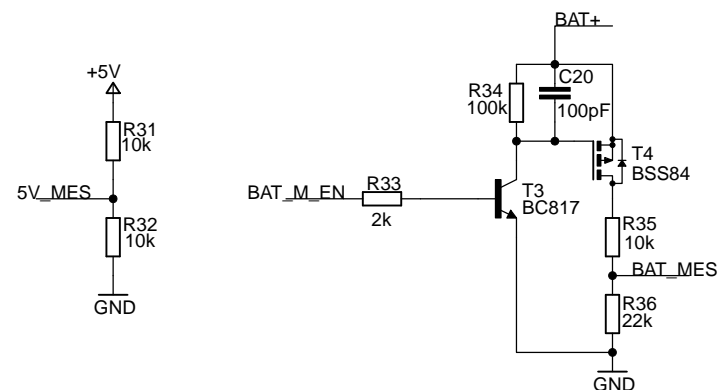
## Accu DC/DC power supply



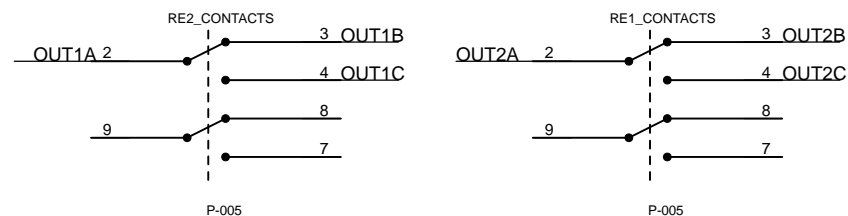
## Connections



## Voltage measurement



## Relay contacts



## Příloha B - Schéma rozšiřující desky

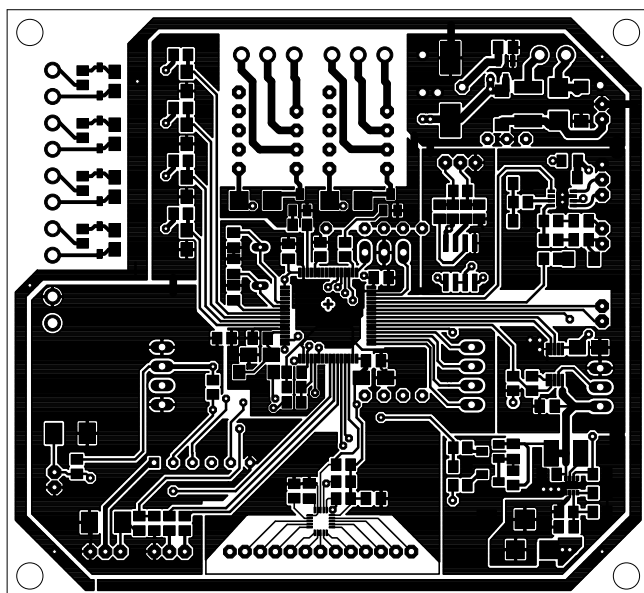
NÁZEV: PRST - rozširujici deska pro MINI2440

Autor: Martin Triska

REV:

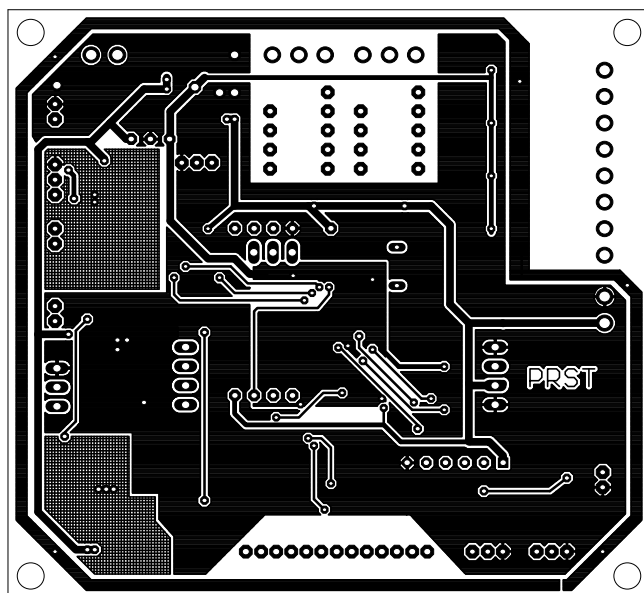
Datum zmeny: 16.05.2013 19:44:16 List: 2/2

## Příloha C - Deska plošných spojů



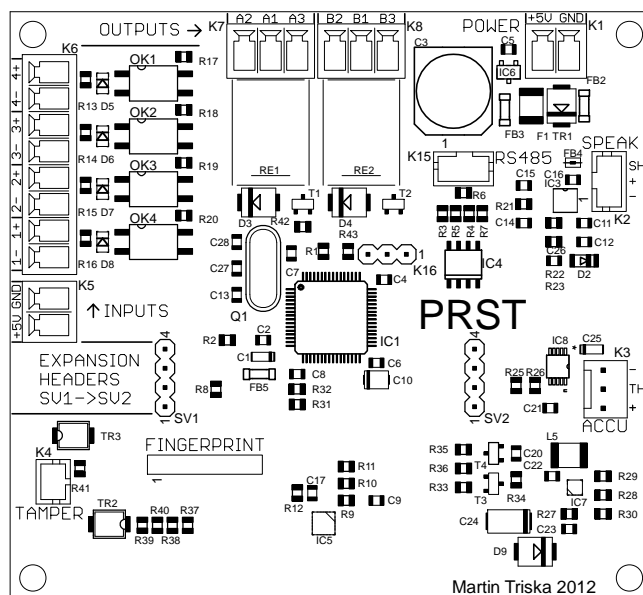
Obrazec desky plošných spojů ze strany TOP

## Příloha C - Deska plošných spojů



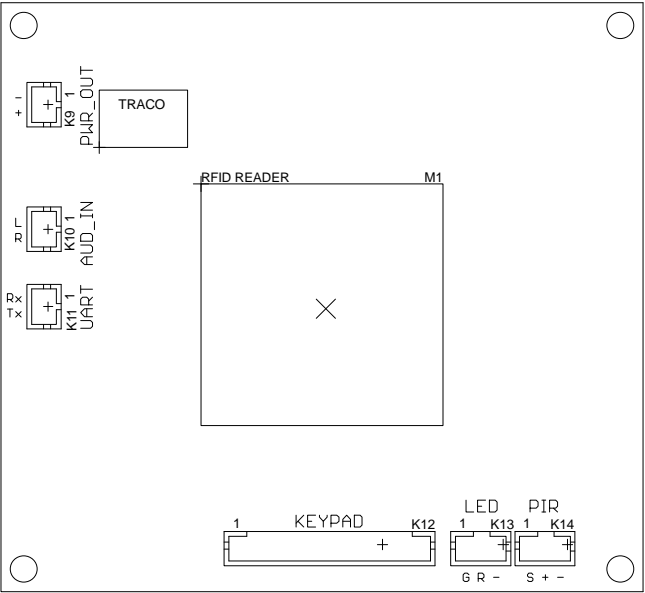
Obrazec desky plošných spojů ze strany BOTTOM - zrcadlený

## Příloha D - Osazovací výkres



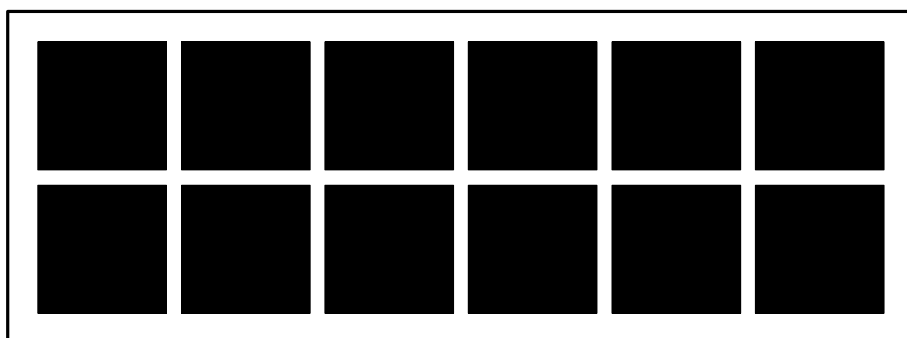
## Osazovací výkres strany TOP

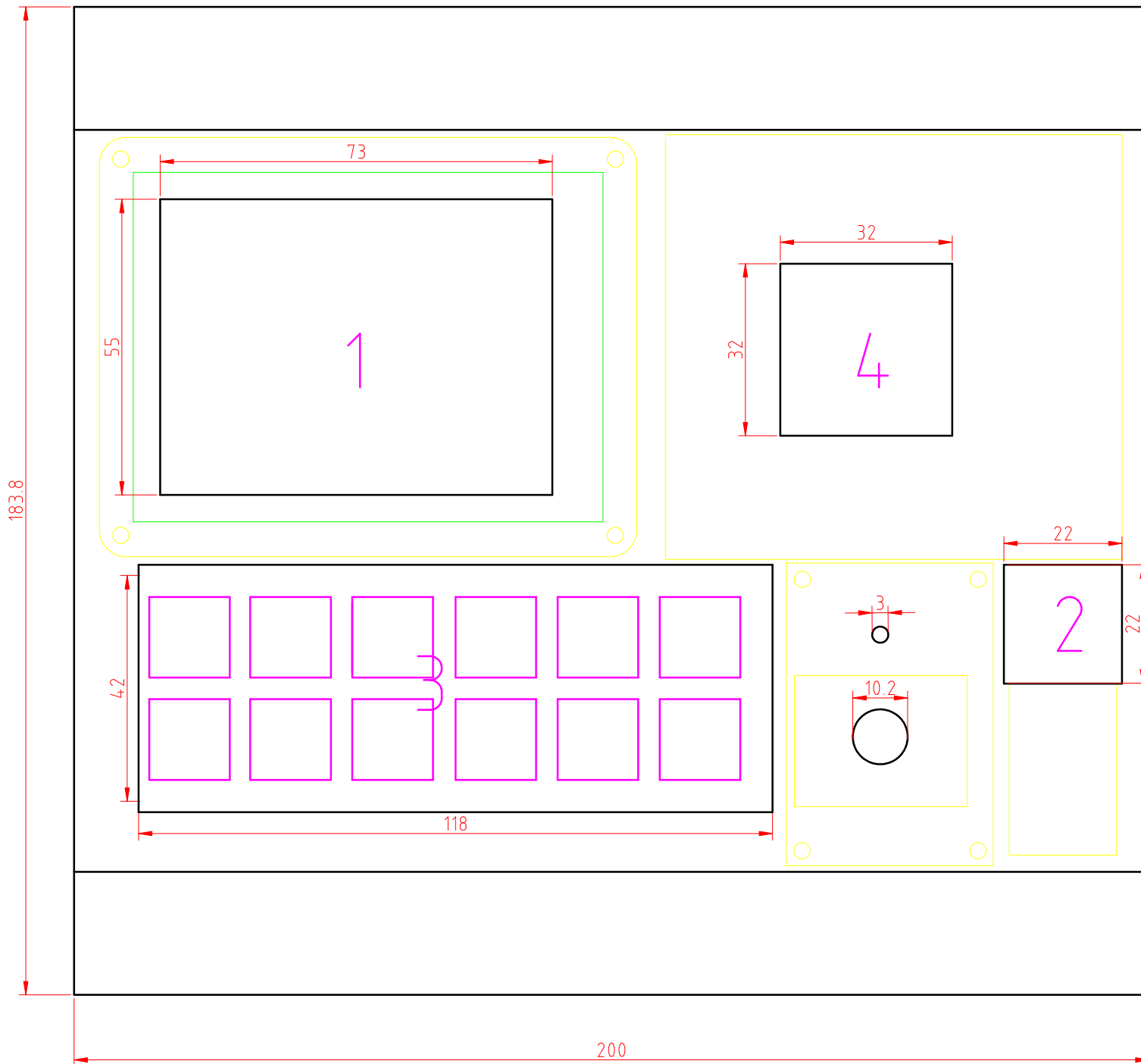
Příloha D - osazovací výkres



Osazovací výkres strany BOTTOM

## Příloha E - Deska plošných spojů kapacitní klávesnice







# PRST Přístupový registrační systém



RFID

Oběd 0	Pauza 1	Dovolená 2 ABC	Lékař 3 DEF	Služebně 4 GHI	
Saldo 5 JKL	Editace 6 MNO	Systém 7 PQRS	Info 8 TUV	9 WXYZ	

STATUS



Docházkový terminál ATS-1

## Příloha H – Fotografie realizovaného přístroje



## Příloha H – Fotografie osazené rozšiřující desky

